



# **PROJETO DE GRADUAÇÃO**

## **Modelagem Numérica de Escoamentos Multifásicos Tridimensionais**

Por: Victor Guinancio e Abicalil

Brasília, 21 de Novembro de 2018



**UNIVERSIDADE DE BRASÍLIA**  
**FACULDADE DE TECNOLOGIA**  
**DEPARTAMENTO DE ENGENHARIA MECÂNICA**

Victor Guinancio e Abicalil

# **Modelagem Numérica de Escoamentos Multifásicos Tridimensionais**

Projeto submetido ao curso de graduação  
em Engenharia Mecânica da Universidade de  
Brasília, como requisito para obtenção do tí-  
tulo de Engenheiro Mecânico.

Universidade de Brasília - UnB

Faculdade de Tecnologia - FT

Orientador: Prof. Dr. Taygoara Felamingo de Oliveira

Brasília, DF

2018

Victor Guinancio e Abicalil

## **Modelagem Numérica de Escoamentos Multifásicos Tridimensionais**

Projeto submetido ao curso de graduação  
em Engenharia Mecânica da Universidade de  
Brasília, como requisito para obtenção do tí-  
tulo de Engenheiro Mecânico.

Trabalho aprovado. Brasília, DF, 21 de Novembro de 2018:

---

**Prof. Dr. Taygoara Felamingo de  
Oliveira**  
Orientador

---

**Prof. Dr. Adriano Possebon**  
Examinador

---

**Prof. Dr. Yuri Dumaresq Sobral**  
Examinador

Brasília, DF  
2018

# Resumo

Este trabalho apresenta um método numérico para a simulação transiente de escoamentos bifásicos incompressíveis tridimensionais, aplicados para os casos de uma cavidade movida pela tampa e de cisalhamento simples. Para isso, utiliza-se o método das diferenças finitas com malha escalonada para a discretização espacial do problema, com o método de Crank-Nicolson para a discretização temporal, resultando em um método de segunda ordem tanto no tempo quanto no espaço. Para resolver as equações de Navier-Stokes, foi utilizado o método de projeção, e, para simular a evolução de uma gota presente no escoamento, foi utilizado o método de Level Set. São apresentados resultados de validação do método para escoamentos monofásicos na cavidade, com  $Re = 100$  e  $Re = 1000$ , com cavidades de diferentes dimensões, bem como resultados de validação do método para escoamentos bifásicos, em cisalhamento simples, para baixos números de  $Re$  e  $Ca$ .

**Palavras-chaves:** Escoamentos Multifásicos, Método de Projeção, Método de Level Set, Método de Diferenças Finitas.

# Abstract

This project presents a numerical method for the simulation of transient tridimensional two-phase incompressible flows, applied to the cases of the lid-driven cavity and of simple shear. The finite difference method with a staggered grid was used for the spacial discretization, and the Crank-Nicolson method was used for the time discretization, resulting in a second order method for both time and space. In order to solve the Navier-Stokes equations, the projection method was used, along with the Level Set method for the simulation of the behaviour of a drop immersed in the flow. Results for monophasic flows in the lid-driven cavity, with varying dimensions, for  $Re = 100$  and  $Re = 1000$ , are presented for validation of the method, as well as validation for biphasic flows, in simple shear, for low  $Re$  and  $Ca$  numbers.

**Key-words:** Multiphase Flow, Projection Method, Level Set Method, Finite Difference Method.

# Lista de ilustrações

Figura 1 – Representação do problema da cavidade . . . . .	14
Figura 2 – Representação do escoamento cisalhante simples . . . . .	16
Figura 3 – Esquematização de um elemento da malha escalonada tridimensional . . . . .	35
Figura 4 – Esquematização de uma malha escalonada bidimensional . . . . .	35
Figura 5 – Análise de ordem da discretização espacial . . . . .	54
Figura 6 – Análise de ordem da discretização temporal . . . . .	54
Figura 7 – Perfil da velocidade $u$ para $x = 0,5$ e $z = 0,5$ . $Re = 100$ , $L_y = 1$ e $L_z = 1$	55
Figura 8 – Perfil da velocidade $v$ para $y = 0,5$ e $z = 0,5$ . $Re = 100$ , $L_y = 1$ e $L_z = 1$	56
Figura 9 – Perfil da velocidade $u$ para $x = 0,5$ e $z = 0,5$ . $Re = 1000$ , $L_y = 1$ e $L_z = 1$ . . . . .	56
Figura 10 – Perfil da velocidade $v$ para $y = 0,5$ e $z = 0,5$ . $Re = 1000$ , $L_y = 1$ e $L_z = 1$ . . . . .	57
Figura 11 – Perfil da velocidade $u$ para $x = 0,5$ e $z = 1$ . $Re = 1000$ , $L_y = 1$ e $L_z = 2$	57
Figura 12 – Perfil da velocidade $v$ para $y = 0,5$ e $z = 1$ . $Re = 1000$ , $L_y = 1$ e $L_z = 2$	58
Figura 13 – Perfil da velocidade $u$ para $x = 0,5$ e $z = 0,5$ . $Re = 1000$ , $L_y = 2$ e $L_z = 1$ . . . . .	58
Figura 14 – Perfis da velocidade $v$ para $y = 0,5$ , $y = 1$ e $y = 1,5$ , com $z = 0,5$ . $Re = 1000$ , $L_y = 2$ e $L_z = 1$ . . . . .	59
Figura 15 – Perfil da velocidade $w$ para $x = 0,5$ e $y = 0,5$ . $Re = 1000$ , $L_y = 1$ . . . . .	59
Figura 16 – Escoamento em uma cavidade cúbica com $Re = 100$ . Plano $z = 0,5$ . . . . .	60
Figura 17 – Linhas de corrente de escoamento em uma cavidade cúbica com $Re = 100$	61
Figura 18 – Escoamento em uma cavidade cúbica com $Re = 1000$ . Plano $z = 0,5$ . . . . .	61
Figura 19 – Linhas de corrente de escoamento em uma cavidade cúbica com $Re = 1000$ . . . . .	61
Figura 20 – Escoamento em uma cavidade com $L_y = 2$ , $L_z = 1$ , $Re = 1000$ . . . . .	62
Figura 21 – Análise de deformação da gota . . . . .	63
Figura 22 – Análise de inclinação da gota . . . . .	63
Figura 23 – Gota imersa em escoamento cisalhante simples, $Re = 0$ e $Ca = 0,02$ . . . . .	65
Figura 24 – Gota imersa em escoamento cisalhante simples, $Re = 0$ e $Ca = 0,1$ . . . . .	65
Figura 25 – Gota imersa em escoamento cisalhante simples, $Re = 0$ e $Ca = 0,2$ . . . . .	65
Figura 26 – Gota imersa em escoamento cisalhante simples, $Re = 0$ e $Ca = 0,4$ . . . . .	66
Figura 27 – Ruptura de gota sob escoamento cisalhante simples. $Re = 0,1$ , $Ca = 0,45$	67
Figura 28 – Análise da perda de volume da gota . . . . .	67
Figura 29 – Comparação de perdas de volume para $Re = 0$ e $Ca = 0,2$ . . . . .	77
Figura 30 – Comparação de ângulos de inclinação para $Re = 0$ e $Ca = 0,2$ . . . . .	77
Figura 31 – Comparação de deformações de Taylor para $Re = 0$ e $Ca = 0,2$ . . . . .	78

Figura 32 – Comparação de perdas de volume para $Re = 0,1$ e $Ca = 0,45$ . . . . .	78
Figura 33 – Comparação de ângulos de inclinação para $Re = 0,1$ e $Ca = 0,45$ . . .	79
Figura 34 – Comparação de deformações de Taylor para $Re = 0,1$ e $Ca = 0,45$ . .	79
Figura 35 – Comparação de perfis da velocidade $u$ para $x = 0,5$ e $z = 0,5$ . $Re = 1000$	80
Figura 36 – Comparação de perfis da velocidade $v$ para $y = 0,5$ e $z = 0,5$ . $Re = 1000$	80
Figura 37 – Comparação de perfis da velocidade $w$ para $x = 0,5$ e $y = 0,5$ . $Re = 1000$	81
Figura 38 – Comparação de perfis da velocidade $u$ para $x = 0,5$ e $z = 0,5$ . $Re =$ 1000, $t = 30$ . . . . .	82
Figura 39 – Comparação de perfis da velocidade $v$ para $y = 0,5$ e $z = 0,5$ . $Re =$ 1000, $t = 30$ . . . . .	82
Figura 40 – Comparação de perfis da velocidade $w$ para $x = 0,5$ e $y = 0,5$ . $Re =$ 1000, $t = 30$ . . . . .	83

# Lista de tabelas

Tabela 1	–	Valores dos <i>ghosts</i> de $u^*$ para o problema da cavidade . . . . .	45
Tabela 2	–	Valores dos <i>ghosts</i> de $u$ e $\chi$ para o problema da cavidade . . . . .	45
Tabela 3	–	Alterações dos parâmetros dos sistemas lineares no contorno, para o caso do problema da cavidade . . . . .	46
Tabela 4	–	Valores dos <i>ghosts</i> de $u^*$ para o caso de cisalhamento simples . . . . .	47
Tabela 5	–	Valores dos <i>ghosts</i> de $u$ e $\chi$ para o caso de cisalhamento simples . . . . .	47
Tabela 6	–	Alterações dos parâmetros dos sistemas lineares no contorno, para o caso de cisalhamento simples . . . . .	47
Tabela 7	–	Algoritmo de Gradientes Conjugados com preconditionamento (SHEW- CHUK et al., 1994) . . . . .	48
Tabela 8	–	Comparação entre formas de cálculo da deformação de Taylor. $Re = 0$	64
Tabela 9	–	Parâmetros dos sistemas lineares na forma do estêncil . . . . .	75



# Lista de abreviaturas e siglas

SSOR	<i>Symmetric Successive Over-Relaxation</i>
GPU	<i>Graphics Processing Unit</i>
CPU	<i>Central Processing Unit</i>
TVD	<i>Total Variation Diminishing</i>
ENO	<i>Essentially Non-Oscillatory</i>
WENO	<i>Weighted Essentially Non-Oscillatory</i>

# Lista de símbolos

$U$	Velocidade da parede
$L_x$	Comprimento do domínio computacional na direção $x$
$L_y$	Comprimento do domínio computacional na direção $y$
$L_z$	Comprimento do domínio computacional na direção $z$
$\gamma$	Coefficiente de tensão superficial
$\dot{\gamma}_c$	Taxa de cisalhamento média
$Re$	Número de Reynolds
$Ca$	Número de capilaridade
$\mathbf{u}$	Vetor velocidade do escoamento
$\mathbf{u}^*$	Vetor velocidade-tentativa do escoamento
$u$	Velocidade do escoamento na direção $x$
$v$	Velocidade do escoamento na direção $y$
$w$	Velocidade do escoamento na direção $z$
$u^*$	Velocidade-tentativa do escoamento na direção $x$
$v^*$	Velocidade-tentativa do escoamento na direção $y$
$w^*$	Velocidade-tentativa do escoamento na direção $z$
$P$	Pressão do escoamento
$t$	Tempo adimensional
$\Delta t$	Intervalo de tempo entre as iterações
$\tau$	Variável de tempo auxiliar para o algoritmo de reinicialização
$\Delta \tau$	Intervalo de tempo auxiliar entre as iterações do algoritmo de reinicialização
$\chi$	Variável auxiliar para o cálculo da pressão
$\Gamma$	Superfície da gota

$V_g$	Volume da gota
$V$	Volume total do domínio computacional
$\kappa$	Curvatura de uma superfície
$\hat{n}$	Vetor normal à uma superfície
$n_x$	Componente na direção $x$ do vetor normal à uma superfície
$n_y$	Componente na direção $y$ do vetor normal à uma superfície
$n_z$	Componente na direção $z$ do vetor normal à uma superfície
$\hat{t}$	Vetor tangencial à uma superfície
$\delta(x)$	Função delta de Dirac
$\delta_{ij}$	Delta de Kronecker
$\epsilon$	Tolerância utilizada para a suavização de funções do Level Set
$\Delta x$	Distância entre pontos adjacentes da malha na direção $x$
$\Delta y$	Distância entre pontos adjacentes da malha na direção $y$
$\Delta z$	Distância entre pontos adjacentes da malha na direção $z$
$NX$	Número de pontos na malha na direção $x$
$NY$	Número de pontos na malha na direção $y$
$NZ$	Número de pontos na malha na direção $z$
$x_c$	Posição inicial do centro da gota na direção $x$
$y_c$	Posição inicial do centro da gota na direção $y$
$z_c$	Posição inicial do centro da gota na direção $z$
$r_g$	Raio inicial da gota
$D_T$	Deformação de Taylor
$L$	Comprimento da gota
$B$	Altura da gota

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>CONTEXTUALIZAÇÃO</b>	<b>13</b>
<b>1.2</b>	<b>O PROBLEMA DA CAVIDADE</b>	<b>14</b>
<b>1.3</b>	<b>ESCOAMENTO CISALHANTE SIMPLES</b>	<b>15</b>
<b>1.4</b>	<b>REVISÃO BIBLIOGRÁFICA</b>	<b>17</b>
<b>2</b>	<b>METODOLOGIA</b>	<b>22</b>
<b>2.1</b>	<b>O MÉTODO DE PROJEÇÃO</b>	<b>22</b>
2.1.1	Método geral	22
2.1.2	Condições de contorno	24
2.1.3	Adaptação para escoamentos de Stokes	25
<b>2.2</b>	<b>INTRODUÇÃO DAS TENSÕES INTERFACIAIS</b>	<b>25</b>
<b>2.3</b>	<b>O MÉTODO DE LEVEL SET</b>	<b>26</b>
<b>2.4</b>	<b>O ALGORITMO DE REINICIALIZAÇÃO</b>	<b>29</b>
<b>2.5</b>	<b>O MÉTODO DE LEVEL SET LOCAL</b>	<b>31</b>
<b>2.6</b>	<b>DISCRETIZAÇÃO TEMPORAL DAS EQUAÇÕES DE TRANSPORTE E REINICIALIZAÇÃO DO LEVEL SET</b>	<b>32</b>
<b>2.7</b>	<b>DISCRETIZAÇÃO EM DIFERENÇAS FINITAS</b>	<b>34</b>
2.7.1	Aspectos gerais	34
2.7.2	Esquemas ENO e WENO	36
2.7.3	Equação da velocidade-tentativa	39
2.7.4	Equação da pressão	40
2.7.5	Equação da velocidade real	41
2.7.6	Equações de transporte e reinicialização do Level Set	41
2.7.7	Condições de contorno para o problema da cavidade	44
2.7.8	Condições de contorno para cisalhamento simples	45
<b>2.8</b>	<b>O MÉTODO DOS GRADIENTES CONJUGADOS</b>	<b>46</b>
<b>2.9</b>	<b>PÓS-PROCESSAMENTO DOS RESULTADOS</b>	<b>50</b>
<b>2.10</b>	<b>DESCRIÇÃO GERAL DO PROGRAMA COMPLETO</b>	<b>51</b>
<b>3</b>	<b>RESULTADOS</b>	<b>53</b>
<b>3.1</b>	<b>ESCOAMENTOS MONOFÁSICOS NA CAVIDADE</b>	<b>53</b>
3.1.1	Verificação de ordem do método	53
3.1.2	Comparação com resultados disponíveis na literatura	55
3.1.3	Análise do escoamento	60
<b>3.2</b>	<b>ESCOAMENTOS BIFÁSICOS DE CISALHAMENTO SIMPLES</b>	<b>62</b>

3.2.1	Comparação com resultados disponíveis na literatura . . . . .	62
3.2.2	Análise do escoamento . . . . .	64
3.2.3	Ruptura de gotas . . . . .	66
4	<b>CONCLUSÕES E TRABALHOS FUTUROS . . . . .</b>	<b>69</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>71</b>
	<b>ANEXO A – PARÂMETROS DOS SISTEMAS LINEARES NA FORMA DE ESTÊNCIL . . . . .</b>	<b>74</b>
	<b>ANEXO B – ANÁLISE DA INFLUÊNCIA DE UTILIZAÇÃO DE VARIÁVEIS DE PRECISÃO SIMPLES OU DUPLA . . . . .</b>	<b>76</b>

# 1 INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

Simulações numéricas de escoamentos multifásicos são aplicáveis em diversas áreas da engenharia, de modo que seu estudo é de grande importância para o desenvolvimento de máquinas e processos mais eficientes. Entre as aplicações para simulações de escoamentos multifásicos, estão a operação de alguns tipos de turbinas, sistemas de injeção de combustível em motores de combustão interna, escoamentos de superfície livre - de extrema importância para a indústria naval, escoamentos de emulsões, fenômenos de cavitação, escoamentos em condensadores e evaporadores, impressoras do tipo *inkjet*, entre outros. Métodos de simulação para escoamentos bifásicos também são de grande importância para a indústria de computação gráfica, porém com uma grande diferença: a única necessidade é a de que o resultado pareça realista, de modo que a acurácia em representar escoamentos reais fica em segundo plano, enquanto a redução do custo computacional é prioridade máxima.

No contexto deste trabalho, o foco está na análise de emulsões, isto é, dispersões de pequenas gotas de um fluido imersas em um fluido matriz, em que os fluidos são imiscíveis. Estas emulsões são formadas por processos de ruptura de gotas, de modo que estas se tornem extremamente pequenas e numerosas, de modo que, em uma escala macroscópica, a emulsão se comporte como um único fluido, de propriedades diferentes das dos dois fluidos que a compõe. Isto pode ser bastante interessante, por exemplo, com a utilização de um fluido magnético em uma das fases, de modo que a emulsão como um todo também apresente propriedades magneto-reológicas, o que apresenta grandes possibilidades para desenvolvimento de tecnologias futuras.

Neste contexto, o objetivo deste trabalho é desenvolver um código computacional para simulação de escoamentos tridimensionais de uma gota de fluido newtoniano, imersa em uma fase contínua de fluido também newtoniano, sujeita a um escoamento incompressível em regime laminar. O método desenvolvido também deve ser capaz de lidar com fenômenos de ruptura e coalescência de gotas. Deste modo, será utilizado o método de Level Set, devido à sua facilidade de implementação e capacidade de lidar com grandes alterações de geometria, incluindo rupturas e coalescência.

Como caso de estudo inicial, com o objetivo de validação do código para escoamentos monofásicos, será utilizado o problema da cavidade tridimensional movida por tampa, devido à simplicidade de geração de malha e aplicação das condições de contorno, bem como à grande disponibilidade de resultados para comparação disponíveis na literatura.

Como caso de validação do código para escoamentos bifásicos, será utilizado o escoamento em cisalhamento simples, também devido à simplicidade de geração de malha e aplicação das condições de contorno, e à grande disponibilidade de resultados para comparação disponíveis na literatura.

Também deseja-se estudar métodos rápidos de solução de sistemas lineares, de modo a diminuir o custo computacional das simulações, tendo em vista o grande número de operações necessárias para a simulação de escoamentos tridimensionais, especialmente para o caso de escoamentos multifásicos.

Por fim, o código deve ser testado e validado, incluindo comparações com resultados disponíveis na literatura, de modo a assegurar o seu funcionamento correto e a qualidade de seus resultados.

## 1.2 O PROBLEMA DA CAVIDADE

O problema inicial a ser estudado neste trabalho é o de uma cavidade tridimensional, movida por uma tampa que se move com velocidade  $U$ . Esta cavidade está representada na Fig. 1, e possui dimensões:

- Comprimento  $L_x$ .
- Altura  $L_y$ .
- Largura  $L_z$ .

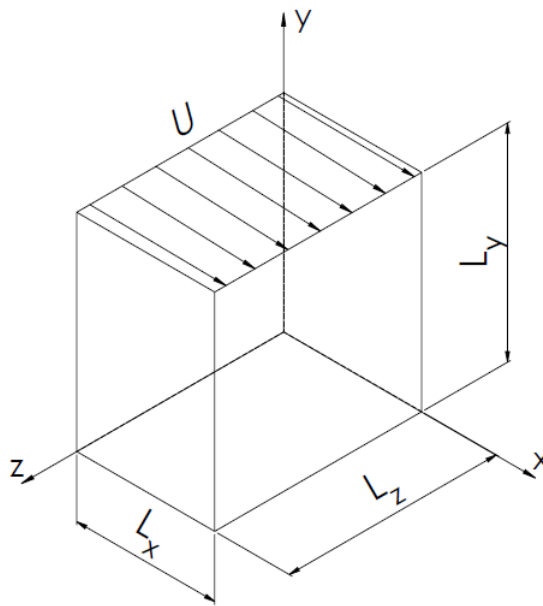


Figura 1 – Representação do problema da cavidade

As condições de contorno aplicadas ao escoamento são de Dirichlet para a velocidade:

- Condição de não-eskorregamento em todas as paredes.
- Condição de impenetrabilidade em todas as paredes.

Além disso, na teoria de camada limite, temos que, em seu interior, variações de pressão significativas ocorrem somente na direção tangencial à parede, com variações na direção normal sendo desprezíveis. Deste modo, em uma região muito próxima à parede, a componente normal do gradiente de pressão será nula, e podemos impor como condição de contorno:

- $\nabla P \cdot \hat{n} = 0$  em todas as paredes.

Para o caso de escoamentos bifásicos, utiliza-se uma gota imersa na cavidade, inicialmente esférica, com coordenadas do centro  $x_c$ ,  $y_c$  e  $z_c$ , e raio inicial  $r_g$ . Os números de  $Re$  e  $Ca$  são definidos como:

$$Re = \frac{\rho U L_x}{\mu}, \quad (1)$$

$$Ca = \frac{\mu U}{\gamma}, \quad (2)$$

em que  $\rho$  é a densidade do fluido matriz (a fase principal do escoamento, dentro da qual podem estar imersas gotas de um outro fluido),  $\mu$  é sua viscosidade dinâmica, e  $\gamma$  é a tensão superficial entre os dois fluidos, para o caso de escoamentos bifásicos.

O objetivo do estudo desta cavidade é validar o código numérico para escoamentos monofásicos, com base em comparações com resultados disponíveis na literatura, além de observar o comportamento de uma gota ao ser transportada pelo escoamento, no caso de escoamentos bifásicos, bem como analisar as influências desta gota sobre o escoamento. Como a principal fonte de dados para validação é o trabalho apresentado por Albensoeder e Kuhlmann (2005), é essencial que a definição de número de Reynolds aqui utilizada seja compatível com a definição utilizada por Albensoeder.

### 1.3 ESCOAMENTO CISALHANTE SIMPLES

Além do problema da cavidade, o outro escoamento a ser estudado neste trabalho é o escoamento cisalhante simples, utilizado para validar o código para o caso de escoamentos bifásicos. Para estabelecer este escoamento, é utilizado um domínio retangular, com uma gota inicialmente esférica posicionada em seu centro, de raio  $r_g$ . Este domínio, representado na Fig. 2, possui dimensões:



- Comprimento  $L_x$ .
- Altura  $L_y$ .
- Largura  $L_z$ .

Nos contornos superior e inferior são empregadas condições de contorno de parede, iguais às utilizadas no problema da cavidade. Isto é:

- Condição de não-eskorregamento nas paredes.
- Condição de impenetrabilidade nas paredes.
- $\nabla P \cdot \hat{n} = 0$  nas paredes.

Ambas as paredes se movem na direção  $x$ , em que a parede superior possui velocidade  $U/2$  e a inferior possui velocidade  $-U/2$ , de modo que a taxa de cisalhamento média é  $\dot{\gamma}_c = U/H$ . Nos demais quatro contornos, são utilizadas condições de contorno periódicas.

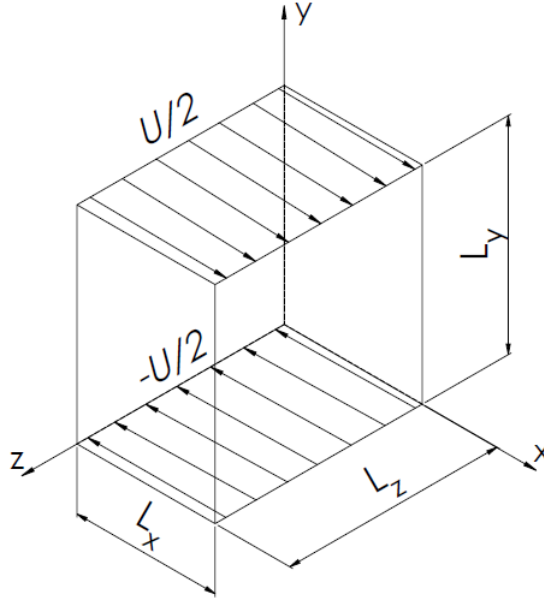


Figura 2 – Representação do escoamento cisalhante simples

Neste caso, os números de  $Re$  e  $Ca$  são definidos como:

$$Re = \frac{\rho \dot{\gamma}_c r_g^2}{\mu}, \quad (3)$$

$$Ca = \frac{r_g \dot{\gamma}_c \mu}{\gamma}, \quad (4)$$

em que  $\rho$  é a densidade do fluido matriz,  $\mu$  é sua viscosidade, e  $\gamma$  é a tensão superficial entre os dois fluidos.

Para simulações de escoamentos cisalhantes simples, o objetivo é observar a deformação da gota, para baixos números de  $Re$  e  $Ca$ , e comparar os resultados obtidos com resultados teóricos, experimentais, e numéricos disponíveis na literatura, de modo a validar o código numérico para escoamentos bifásicos.

## 1.4 REVISÃO BIBLIOGRÁFICA

Escoamentos multifásicos são um tema de grande interesse para a mecânica dos fluidos. Entretanto, a análise numérica destes é relativamente recente, tendo em vista os grandes custos computacionais associados à simulações destes escoamentos. Análises teóricas e experimentais, entretanto, começaram a ser realizadas por Taylor já na década de 1930.

Inicialmente, Taylor (1932) propõe um modelo matemático para calcular a influência de gotas de um fluido imersas em um outro fluido, com base na hipótese de que estas gotas permaneçam aproximadamente esféricas. Com isto, Taylor apresenta um modelo matemático para determinar a viscosidade equivalente da emulsão. Apesar desta limitação de forma, os resultados apresentados por Taylor são úteis ao se trabalhar com gotas pequenas e com fortes tensões superficiais, já que nestes casos as deformações da gota são pequenas.

Posteriormente, Taylor (1934) trouxe a análise do comportamento de escoamentos bifásicos para o âmbito experimental, observando as deformações de gotas de diferentes fluidos quando imersas em escoamentos. Neste trabalho, Taylor também apresenta uma teoria matemática para representar a deformação das gotas, variando linearmente em função do número de capilaridade, bem como prevê uma inclinação de  $45^\circ$  entre o eixo longo da gota e a direção principal do escoamento, para o caso de cisalhamento simples. Além disso, Taylor também introduz uma forma de medição da deformação das gotas, amplamente utilizada, e denominada como deformação de Taylor, além de descrever fenômenos de ruptura de gotas. Outras análises experimentais também incluem as realizadas por Rumscheidt e Mason (1961), Torza, Cox e Mason (1972), Bentley e Leal (1986), entre outros.

Já na área de simulações numéricas de escoamentos bifásicos, o problema pode, essencialmente, ser separado em duas partes: a modelagem do escoamento, e o rastreamento e transporte da interface entre os dois fluidos. Para ambas as partes, há uma grande quantidade de trabalhos desenvolvidos, construindo diversos métodos diferentes de simulação, cada um com suas vantagens e desvantagens.

Para a questão de rastreamento da interface, os métodos podem ser divididos em dois grupos: métodos de superfícies explícitas, nos quais a posição da interface entre os fluidos é calculada de forma explícita, e métodos de superfícies implícitas, em que a posição da interface não é calculada de forma explícita, e precisa ser determinada de forma implícita. Métodos explícitos incluem métodos de integral de contorno (*Boundary Integral Method*), bem como métodos de volumes de fluido (*Volume of Fluid*). Os métodos de integral de contorno definem a interface de forma paramétrica, com a malha computacional disposta na superfície da gota, de modo que ocorre um rastreamento da superfície de forma explícita. Deste modo, como os elementos de malha estão dispostos na superfície, este método apresenta uma enorme dificuldade em lidar com grandes variações de geometria, como ruptura e coalescência de gotas. Já métodos de volumes de fluido, inicialmente propostos por Hirt e Nichols (1981), discretizam o interior da gota em elementos que contêm frações de seu volume, de modo que estes métodos rastreiam de forma explícita o fluido contido no interior da gota. Estes métodos são capazes de lidar com fenômenos de ruptura e coalescência de gotas de forma simples, mas apresentam uma grande dificuldade na determinação da curvatura da interface.

Já para o caso de métodos implícitos, o mais proeminente é o método de Level Set (*Level Set Method*), proposto por Osher e Sethian (1988). Este método consiste em criar um campo escalar, com a interface definida como a curva de nível para um determinado valor desta função, fazendo com que a interface seja rastreada de forma implícita. Este método é capaz de lidar com grandes alterações de geometria, como ruptura e coalescência de gotas, de forma automática, bem como permite que a curvatura da superfície seja determinada de forma bastante simples. A grande desvantagem deste método, quando comparado aos outros apresentados, é que ele não garante a conservação de massa da gota, de modo que ele pode não se mostrar vantajoso para simulações correspondentes a fenômenos de longa duração. Uma boa revisão acerca das formas de implementação de métodos de Level Set, bem como algumas de suas aplicações, é a apresentada por Osher e Fedkiw (2001), e um exemplo recente de novos desenvolvimentos no método de Level Set é o método *narrow-band gradient-augmented* apresentado por Lee, Dolbow e Mucha (2014).

Exemplos da utilização de métodos de integral de contorno para a simulação de deformações de gotas em escoamentos cisalhantes incluem os trabalhos desenvolvidos por Rallison (1981), e Kennedy, Pozrikidis e Skalak (1994), e um exemplo da utilização de métodos de volume de fluido para este mesmo caso é o trabalho desenvolvido por Li, Renardy e Renardy (2000).

Para a questão de modelagem do escoamento, a utilização de métodos de solução das equações de Navier-Stokes se faz necessária, para permitir a simulação dos escoamentos formados tanto no interior quanto no exterior das gotas. Neste sentido, um dos

métodos mais proeminentes para simulações transientes de escoamentos incompressíveis são os métodos de projeção, como por exemplo o apresentado por Kim e Moin (1985). Um trabalho de grande relevância no contexto de métodos de projeção é o de Brown, Cortez e Minion (2001), que condensa diferentes versões de métodos de projeção para modelagem de escoamentos incompressíveis, listando suas características e formas de implementação, de modo que este trabalho se caracteriza como uma boa fonte de referência inicial para estudos nesta área.

Para realizar a simulação de escoamentos bifásicos, Sussman, Smereka e Osher (1994) apresentam uma metodologia construída com base em um método de projeção para modelar o escoamento, acoplada a um método de Level Set para rastreamento da interface, com o objetivo de simular o comportamento de bolhas de ar imersas em água, e gotas de água imersas em ar. Neste trabalho, Sussman apresenta um algoritmo de reinicialização da função do Level Set, de modo a mantê-la como uma função distância, o que torna o método de Level Set capaz de lidar com grandes razões de viscosidade e densidade entre os fluidos, bem como com grandes tensões interfaciais. Este algoritmo consiste em encontrar a solução para regime permanente de uma equação, conhecida como equação de reinicialização, aplicada à função Level Set, isto é, este método altera a função Level Set até que ela esteja suficientemente próxima de uma função distância. Min (2010) apresenta uma análise detalhada de diferentes abordagens de implementação deste tipo de algoritmo de reinicialização. Como uma alternativa a esta forma de reinicialização, existem métodos conhecidos como *Fast Marching Methods*, baseados no algoritmo desenvolvido por Tsitsiklis (1995), que constroem de forma direta uma função distância. Também existe uma formulação variacional do método de Level Set, proposta por Li et al. (2005), que dispensa a utilização de algoritmos de reinicialização.

Este mesmo problema de gotas de fluidos com densidades e viscosidades diferentes também é abordado por Chang et al. (1996), que introduz uma metodologia de cálculo baseada em uma formulação de vorticidade-função corrente, além de demonstrar uma forma de utilização da função Level Set para representar escoamentos com três fluidos distintos.

Posteriormente, Sussman et al. (1998) introduz uma metodologia de correção do algoritmo de reinicialização, de modo a forçar uma conservação de massa das gotas. Deste modo, é possível reduzir consideravelmente as perdas de massa associadas à erros numéricos do algoritmo de reinicialização. Esta mesma metodologia também é apresentada com mais detalhes por Sussman e Fatemi (1999). Alternativamente, uma melhor conservação de massa também pode ser obtida com um método híbrido, descrito por Enright et al. (2002), e denominado de *Particle Level Set*. Este método consiste em popular a região próxima à interface com partículas marcadoras, movidas livremente por advecção e rastreadas de forma explícita, ao passo em que a interface é rastreada e movida com

a utilização de um método de Level Set. Entretanto, caso uma destas partículas cruze a interface, o método interpreta isso como um erro do método de Level Set, e reconstrói localmente a interface, com base nas posições das partículas marcadoras. Deste modo, este método híbrido entre rastreamento implícito e explícito da interface consegue lidar facilmente com grandes alterações de geometria, e permite que a curvatura da interface seja calculada de forma fácil, ao passo em que também apresenta uma boa conservação de massa.

Com o intuito de diminuir o custo computacional associado ao método de Level Set, Adalsteinsson e Sethian (1995) propõem um método em que a função distância do Level Set é calculada somente em uma região próxima à interface entre as duas fases do escoamento, construindo o que é conhecido como método de Level Set Local. Além da redução do custo computacional, este método também é adequado para resolver problemas em que o campo de velocidades não esteja definido em regiões distantes da interface, como por exemplo em problemas de crescimento de cristais.

Uma outra metodologia para a utilização de um método de Level Set local, consideravelmente mais simples, é a proposta por Peng et al. (1999), juntamente com melhorias para o algoritmo iterativo de reinicialização descrito por Sussman, Smereka e Osher (1994).

Em todos os trabalhos previamente citados, a função de Level Set é tomada como sendo uma função distância, e a interface é definida como a região de valor zero desta função. Entretanto, é possível realizar simulações de escoamentos multifásicos com funções diferentes para o método de Level Set. No trabalho desenvolvido por Olsson e Kreiss (2005), por exemplo, é utilizada uma função Heaviside suavizada, e a interface é definida como a região de valor 0,5 desta função.

No contexto de simulação de escoamentos monofásicos, a serem estudados neste trabalho de forma a simular o problema da cavidade, Albensoeder e Kuhlmann (2005) apresentam uma grande quantidade de resultados para escoamentos em cavidades tridimensionais com diferentes dimensões, e variados números de Reynolds. Como este trabalho utiliza métodos espectrais, de alta ordem, os resultados obtidos são de grande precisão, de modo que são bons valores de referência para validação de métodos numéricos. Um outro trabalho que aborda o problema da cavidade, porém para o caso bidimensional, é o desenvolvido por Ghia, Ghia e Shin (1982), que utiliza métodos de *multigrid* para encontrar soluções para escoamentos com números de Reynolds de até 10000. Métodos de *multigrid* consistem em solucionar o escoamento em malhas de refinamento variável, de modo que são utilizadas malhas mais grossas em regiões de solução mais simples, com um refinamento da malha sendo realizado somente em regiões de solução mais complexa. Deste modo, métodos de *multigrid* permitem uma grande redução no custo computacional, sem uma perda significativa de precisão, porém sua implementação é bastante complexa.

Um outro trabalho que aborda a utilização de métodos de *multigrid*, além do método dos gradientes conjugados, aplicados às equações de Navier-Stokes para escoamentos incompressíveis, porém de forma pouco ortodoxa, é o desenvolvido por Bolz et al. (2003). Neste trabalho, é apresentado um método em que são utilizadas GPUs para realizar as operações computacionais, ao invés de CPUs, como é usual. Neste trabalho, foram realizadas simulações tanto em CPU quanto em GPU, com a GPU apresentando um melhor desempenho. Deste modo, dependendo da disponibilidade de *hardware*, a utilização de GPUs para computação pode ser bastante vantajosa, o que pode ser realizado com a utilização da plataforma CUDA, da Nvidia, ou com a plataforma OpenCL.

## 2 METODOLOGIA

### 2.1 O MÉTODO DE PROJEÇÃO

#### 2.1.1 Método geral

As equações que governam o problema a ser resolvido, desconsiderando-se as tensões superficiais na interface, são as equações de Navier-Stokes para escoamentos incompressíveis, com densidades e viscosidades constantes. Para um campo de velocidades  $\mathbf{u} = (u, v, w)$ , em que  $u$ ,  $v$  e  $w$  são as componentes relativas às direções  $x$ ,  $y$ , e  $z$ , respectivamente, e um campo de pressão  $P$ , a forma adimensional destas equações pode ser escrita como

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla P + \frac{1}{Re} \nabla^2 \mathbf{u}, \quad (5)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (6)$$

em que  $Re$  é o número de Reynolds,  $t$  é o tempo, e  $\nabla$  pode ser interpretado como um vetor de derivadas parciais a serem aplicadas a alguma função, isto é,  $\nabla = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$ . Deste modo,  $\nabla \mathbf{u}$  é o gradiente de  $\mathbf{u}$ ,  $\nabla^2 \mathbf{u}$  é o laplaciano de  $\mathbf{u}$ , e  $\nabla \cdot \mathbf{u}$  é o divergente de  $\mathbf{u}$ .

Uma das grandes dificuldades em resolver estas equações está no fato de que não há uma equação explícita para a pressão. Para contornar este problema, uma das alternativas para solução de escoamentos transientes é a utilização de métodos de projeção. Estes métodos consistem em, para cada avanço no tempo, evoluir o campo de velocidades para um campo de velocidades-tentativa, definido como  $\mathbf{u}^* = (u^*, v^*, w^*)$ , que posteriormente é corrigido de modo a impor a condição de incompressibilidade.

Neste trabalho, o método de projeção utilizado é baseado no que foi proposto por Kim e Moin (1985). Para uma discretização temporal pelo método de Euler explícito, o método de projeção utilizado pode ser representado, de forma simplificada, da seguinte forma. Inicialmente, calcula-se um campo de velocidades-tentativa, com base nas equações de Navier-Stokes, porém desconsiderando-se os efeitos do campo de pressão:

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\mathbf{u}^n \cdot \nabla \mathbf{u}^n + \frac{1}{Re} \nabla^2 \mathbf{u}^n. \quad (7)$$

O campo de velocidades reais, então, pode ser calculado como

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla P. \quad (8)$$

Nota-se que o campo de velocidades-tentativa  $\mathbf{u}^*$  está situado no instante  $n + 1$ .

Como esta derivada temporal não é centrada, isto é uma aproximação de primeira ordem no tempo. Nota-se que, ao somar as Eqs. 7 e 8, e tomando o limite com  $\Delta t \rightarrow 0$ , obtêm-se a Eq. 5, demonstrando que o método é consistente.

Para impor a condição de incompressibilidade ao escoamento, toma-se o divergente da Eq. 8, e, com a utilização da Eq. 6, obtêm-se

$$\nabla^2 P = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^*. \quad (9)$$

Nota-se que, com a utilização desta equação para a pressão, a condição de incompressibilidade é imposta, e recalculada, em todos os passos de tempo, de modo que não há um acúmulo de erros numéricos sobre a condição de incompressibilidade, como pode ocorrer em outros métodos propostos para a solução das equações de Navier-Stokes.

Para obter um método de segunda ordem no tempo, é necessário que a derivada temporal seja centrada. Deste modo, avalia-se todos os termos do lado direito da Eq. 7 no instante  $n + \frac{1}{2}$ , resultando em um método de Crank-Nicolson. Conforme descrito por Kim e Moin (1985) e Brown, Cortez e Minion (2001), a Eq. 7 é modificada para

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -[\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} + \frac{1}{2Re} \nabla^2 (\mathbf{u}^* + \mathbf{u}^n). \quad (10)$$

Como o termo  $[\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}}$  ainda é desconhecido, é necessário estimá-lo por extrapolação. Para isto, utiliza-se o método de Adams-Bashforth, conforme apresentado por Kim e Moin (1985), de modo que

$$[\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} = \frac{3}{2} [\mathbf{u} \cdot \nabla \mathbf{u}]^n - \frac{1}{2} [\mathbf{u} \cdot \nabla \mathbf{u}]^{n-1}. \quad (11)$$

Como este termo é convectivo, a utilização de esquemas de *upwinding* é recomendada para melhorar a estabilidade do método numérico, em especial para o caso de altos números de Reynolds, em que a estabilização promovida pelas tensões viscosas se torna pequena. A implementação destes esquemas é simples, tendo em vista que este termo é calculado de forma explícita.

Nota-se que, ao somar as Eqs. 10 e 8, e tomar o limite com  $\Delta t \rightarrow 0$ , não se recupera as equações de Navier-Stokes, devido à presença do termo com  $\mathbf{u}^*$  do lado direito da equação. Deste modo, para que o método se torne consistente, é necessário realizar algumas modificações, com a introdução de uma variável auxiliar de pressão  $\chi$ . As Eqs. 9 e 8 são então modificadas para

$$\nabla^2 \chi^{n+1} = \frac{1}{\Delta t} \nabla \cdot \mathbf{u}^*, \quad (12)$$

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \Delta t \nabla \chi^{n+1}. \quad (13)$$



Por fim, deseja-se determinar uma descrição para a pressão real  $P$  que torne o método consistente. Para isto, inicialmente toma-se a soma das Eqs. 10 e 13, e, substituindo-se o valor de  $u^*$ , obtêm-se

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = -\nabla \chi^{n+1} - [\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} + \frac{1}{2Re} \nabla^2 (\mathbf{u}^{n+1} + \mathbf{u}^n + \Delta t \nabla \chi^{n+1}). \quad (14)$$

Reorganizando a Eq. 14, e comutando os operadores gradiente e laplaciano, têm-se que

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} = -\nabla (\chi^{n+1} + \frac{\Delta t}{2Re} \nabla^2 \chi^{n+1}) - [\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} + \frac{1}{2Re} \nabla^2 (\mathbf{u}^{n+1} + \mathbf{u}^n). \quad (15)$$

Deste modo, podemos definir a pressão como

$$P^{n+\frac{1}{2}} = \chi^{n+1} + \frac{\Delta t}{2Re} \nabla^2 \chi^{n+1}, \quad (16)$$

ou, alternativamente, usando a Eq. 12,

$$P^{n+\frac{1}{2}} = \chi^{n+1} - \frac{1}{2Re} \nabla \cdot \mathbf{u}^*. \quad (17)$$

Nota-se que, tomando o limite com  $\Delta t \rightarrow 0$  da Eq. 15, e utilizando esta definição para a pressão, obtêm-se as equações de Navier-Stokes, confirmando que esta definição da pressão real  $P$  torna o método consistente. Como demonstrado por Brown, Cortez e Minion (2001), esta definição também é consistente com o método de Crank-Nicolson utilizado, pois está situada no instante  $n + \frac{1}{2}$  junto aos outros termos. Deste modo, esta é uma aproximação de segunda ordem para a pressão no instante  $t^{n+\frac{1}{2}}$ .

### 2.1.2 Condições de contorno

Para  $\chi$ , as condições de contorno se mantêm iguais às condições de contorno para  $P$ :

$$\nabla \chi \cdot \hat{n} = 0, \text{ nas paredes}, \quad (18)$$

Para aplicar as condições de contorno da velocidade, deve-se atentar ao fato de que  $\mathbf{u}^*$  e  $\mathbf{u}$  são diferentes, de modo que as condições de contorno definidas para  $\mathbf{u}$  não podem ser aplicadas diretamente para  $\mathbf{u}^*$ . Deste modo, utiliza-se a Eq. 13 para se correlacionar as condições de contorno de  $\mathbf{u}$  e  $\mathbf{u}^*$ , e determinar as condições de contorno a serem aplicadas à  $\mathbf{u}^*$ . Na direção normal à parede:

$$\mathbf{u} \cdot \hat{n} = \mathbf{u}^* \cdot \hat{n} - \Delta t \nabla \chi \cdot \hat{n}. \quad (19)$$

Utilizando-se a Eq. 18, temos que

$$\mathbf{u}^* \cdot \hat{n} = \mathbf{u} \cdot \hat{n}, \quad (20)$$

isto é, as condições de contorno conhecidas para  $\mathbf{u}$  podem ser diretamente aplicadas à  $\mathbf{u}^*$  quando a componente da velocidade for normal à parede. Já na direção tangencial à parede, temos que

$$\mathbf{u}^* \cdot \hat{t} = \mathbf{u} \cdot \hat{t} + \Delta t \nabla \chi \cdot \hat{t}, \quad (21)$$

de modo que as condições de contorno conhecidas para  $\mathbf{u}$  devem ser modificadas antes de serem aplicadas à  $\mathbf{u}^*$ . Como  $u^*$  está no instante de tempo  $n + 1$ , e o valor de  $\chi^{n+1}$  ainda não é conhecido, é necessário estimá-lo por extrapolação, o que é feito com a utilização do método de Adams-Bashforth, conforme descrito por Weinan, Liu et al. (2003):

$$\chi^{n+1} = 2\chi^n - \chi^{n-1}. \quad (22)$$

### 2.1.3 Adaptação para escoamentos de Stokes

Para a realização de algumas simulações, em que os efeitos dos termos de inércia das equações de Navier-Stokes não são especialmente relevantes, pode-se desejar realizar simulações para o chamado escoamento de Stokes, também conhecido como *creeping flow*. Estes escoamentos permitem uma simplificação das equações governantes, com a eliminação dos termos relativos à inércia e à derivadas temporais, e são caracterizados por uma elevada viscosidade do fluido, baixas velocidades, e pequenas escalas de comprimento, de modo que  $Re \rightarrow 0$ .

Nota-se que, ao tentar simplesmente introduzir  $Re = 0$  na Eq. 10, isto criaria uma singularidade. Entretanto, é fácil notar que  $Re = 0$  significa apenas que o termo relativo às tensões viscosas é predominante, e os termos relativos à inércia são desprezíveis. Deste modo, uma alternativa para o cálculo de escoamentos de Stokes é simplesmente considerar o termo  $\mathbf{u} \cdot \nabla \mathbf{u}$  como sendo nulo, o que faz com que, em regime permanente, quando as derivadas temporais se tornam zero, a solução se torne equivalente à solução para escoamento de Stokes.

## 2.2 INTRODUÇÃO DAS TENSÕES INTERFACIAIS

Na região da interface entre dois fluidos imiscíveis, sempre que esta interface for curva, ocorrerá um salto de pressão entre os dois fluidos, correspondente à existência da tensão interfacial. Este salto de pressão é descrito pela equação de Young-Laplace, na forma:

$$\Delta P_{\Gamma} = -\gamma \kappa, \quad (23)$$

em que  $\gamma$  é o coeficiente de tensão superficial entre os dois fluidos, e  $\kappa$  é a curvatura da superfície. Para introduzir este termo nas equações de Navier-Stokes, entretanto, é necessário antes determinar a posição em que este salto de tensões ocorre, e em qual direção ele ocorre. É fácil notar que a posição é a superfície definida na interface entre

os dois fluidos, porém a determinação matemática desta posição não é trivial. Para isto, pode-se definir um gradiente de pressão na interface, com a utilização de uma função delta de Dirac, de forma que  $|\nabla P_\Gamma| = -\gamma\kappa\delta(\mathbf{x} - \mathbf{x}_\Gamma)$ , em que  $\mathbf{x}_\Gamma$  representa a posição da interface. Com as propriedades da função delta de Dirac, nota-se que este gradiente de pressão será nulo em todo o domínio, exceto na interface entre os fluidos, e que o salto de pressão total é idêntico ao definido pela equação de Young-Laplace, tendo em vista que a integral do delta de Dirac em um domínio que inclua o ponto  $\mathbf{x} = \mathbf{x}_\Gamma$  é igual a 1. Deste modo, resta apenas determinar a direção deste gradiente de pressão, que claramente deve ser uma direção normal à superfície. Com isso, pode-se definir um gradiente de pressão gerado pelas tensões interfaciais, descrito como:

$$\nabla P_\Gamma = -\gamma\kappa\delta(\mathbf{x} - \mathbf{x}_\Gamma)\hat{n}. \quad (24)$$

Este gradiente de pressão pode, então, ser inserido nas equações de Navier-Stokes. No método de projeção empregado, este termo é incluído no lado direito da Eq. 10. Com o termo relativo às tensões interfaciais já adimensionalizado, e definido no instante  $n + \frac{1}{2}$  para estar de acordo com o método de Crank-Nicolson, esta equação é reescrita na forma

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -[\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} + \frac{1}{2Re} \nabla^2(\mathbf{u}^* + \mathbf{u}^n) - \frac{1}{ReCa} [\kappa\delta(\mathbf{x} - \mathbf{x}_\Gamma)\hat{n}]^{n+\frac{1}{2}}, \quad (25)$$

em que  $Ca$  é o número de capilaridade.

## 2.3 O MÉTODO DE LEVEL SET

Com o termo devido às tensões interfaciais devidamente introduzido nas equações de Navier-Stokes, resta o problema de definir a posição da interface, bem como a forma com a qual ela é transportada, e formas de calcular sua curvatura e vetores normais à ela. Isto pode ser feito com a utilização de diversos métodos diferentes, entre eles o Level Set, que é o método utilizado neste trabalho. Este método apresenta diversas vantagens, como sua relativa facilidade de implementação e cálculo de vetores normais e curvaturas, além de sua capacidade de lidar com fenômenos de ruptura e coalescência de gotas de forma direta.

O método de Level Set consiste em criar uma variável  $\phi$ , que inicialmente é definida como uma função distância em relação à interface, assumindo valores positivos para o lado de fora da gota, e valores negativos para o lado de dentro da gota. Deste modo, a interface é definida pelo conjunto de pontos em que  $\phi = 0$ . Esta função distância é definida de forma que, para cada ponto, o valor desta função é a menor distância possível entre este ponto e a interface.

De acordo com Sussman et al. (1998), para que a interface seja transportada corretamente pelo escoamento, têm-se que

$$\frac{D\phi}{Dt} = 0, \quad (26)$$

ou, alternativamente,

$$\frac{\partial \phi}{\partial t} = -\mathbf{u} \cdot \nabla \phi, \quad (27)$$

isto é,  $\phi$  é transportado com uma equação de advecção, de modo que a utilização de esquemas de *upwinding* é essencial. Como o campo de velocidades  $\mathbf{u}$  já é previamente determinado, a implementação destes esquemas pode ser realizada de forma bastante direta.

Vale notar que, apesar de a função  $\phi$  inicialmente ser uma função distância, a Eq. 27 não garante que ela se mantenha como tal. Deste modo, a utilização de um algoritmo de reinicialização se faz necessária, para manter  $\phi$  suficientemente próxima de uma função distância. Isto faz com que a espessura da interface, definida a seguir, seja uniforme, resultando em uma maior estabilidade do método. Além disso, algumas simplificações podem ser feitas ao se assumir que  $|\nabla \phi| = 1$ , as quais serão apresentadas a seguir. Em todo o resto deste trabalho, as definições assumirão que  $|\nabla \phi| = 1$ .

É evidente que os valores de  $|\nabla \phi|$  devem ser constantemente monitorados na região próxima à interface, de modo a garantir que eles estejam sempre suficientemente próximos de 1, sob pena de introduzir erros nos cálculos das tensões interfaciais.

Para uma interface definida pelo método de Level Set, é fácil notar que um vetor normal possuirá a mesma direção que  $\nabla \phi$ . Isto ocorre porque, como a superfície da interface é definida como a região em que  $\phi = 0$ , temos que  $\nabla \phi \cdot \hat{t} = 0$ , de modo que  $\nabla \phi$  é ortogonal à interface e, portanto, normal à ela. Deste modo, como o vetor normal deve ser unitário, temos que

$$\hat{n} = \frac{\nabla \phi}{|\nabla \phi|}. \quad (28)$$

Para o caso de  $|\nabla \phi| = 1$ , o vetor normal é definido simplesmente como  $\hat{n} = \nabla \phi$ . Também podemos definir a curvatura como uma variação do vetor normal, de forma que

$$\kappa = \nabla \cdot \hat{n}, \quad (29)$$

ou, para o caso de  $|\nabla \phi| = 1$ ,  $\kappa = \nabla \cdot (\nabla \phi)$ .

Por fim, também podemos definir a função delta de Dirac em função de  $\phi$ . Para tal, primeiramente é necessário introduzir a função Heaviside,  $H(\phi)$ , definida como sendo igual a zero no interior da gota ( $\phi \leq 0$ ), e igual a um no exterior da gota ( $\phi > 0$ ). Com estas propriedades, pode-se observar que a função delta de Dirac é a derivada direcional da função Heaviside. Como a interface é definida pela superfície em que  $\phi = 0$ , e também

como a superfície em que  $\mathbf{x} - \mathbf{x}_\Gamma = 0$ , temos que

$$\delta(\mathbf{x} - \mathbf{x}_\Gamma) = \nabla H(\phi(\mathbf{x})) \cdot \hat{n}. \quad (30)$$

Utilizando a regra da cadeia, temos que:

$$\delta(\mathbf{x} - \mathbf{x}_\Gamma) = H'(\phi(\mathbf{x})) \nabla \phi(\mathbf{x}) \cdot \hat{n} = H'(\phi(\mathbf{x})) \nabla \phi(\mathbf{x}) \cdot \frac{\nabla \phi(\mathbf{x})}{|\nabla \phi(\mathbf{x})|}. \quad (31)$$

Deste modo, como  $H'(\phi) = \delta(\phi)$ , e  $\nabla \phi \cdot \nabla \phi = |\nabla \phi|^2$ , temos a seguinte definição, apresentada por Osher e Fedkiw (2006):

$$\delta(\mathbf{x} - \mathbf{x}_\Gamma) = \delta(\phi) |\nabla \phi|, \quad (32)$$

ou, para o caso de  $|\nabla \phi| = 1$ :

$$\delta(\mathbf{x} - \mathbf{x}_\Gamma) = \delta(\phi). \quad (33)$$

Deste modo, observando que  $|\nabla \phi| \hat{n} = \nabla \phi$  (Eq. 28), e utilizando a Eq. 32, podemos reescrever a Eq. 25 de acordo com as definições do método de Level Set, conforme apresentado por Sussman et al. (1998). Esta equação é definida como

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -[\mathbf{u} \cdot \nabla \mathbf{u}]^{n+\frac{1}{2}} + \frac{1}{2Re} \nabla^2 (\mathbf{u}^* + \mathbf{u}^n) - \frac{1}{ReCa} [\kappa \delta(\phi) \nabla \phi]^{n+\frac{1}{2}}, \quad (34)$$

em que a curvatura  $\kappa$  é calculada com a utilização da Eq. 29. Nota-se que  $\phi^{n+\frac{1}{2}}$  ainda é desconhecido e precisa ser estimado por extrapolação, o que é feito pelo método de Adams-Bashforth:

$$\phi^{n+\frac{1}{2}} = \frac{3}{2} \phi^n - \frac{1}{2} \phi^{n-1}. \quad (35)$$

Nota-se que a Eq.34 é válida somente para valores de densidade e viscosidade constantes. Deste modo, o método numérico aqui apresentado é limitado a casos em que a densidade e viscosidade de ambos os fluidos sejam iguais. Para a simulação de casos em que as densidades e viscosidades dos fluidos são diferentes, seria necessário que a metodologia utilizada permitisse densidades e viscosidades variáveis, o que acarretaria em grandes mudanças no método de projeção utilizado.

Para o caso de simulações com dados discretos, em que os valores de  $\phi$  são definidos apenas em pontos de uma malha, é evidente que a definição tradicional da função delta de Dirac não é adequada, pois ela falharia em capturar a interface adequadamente. Deste modo, é utilizada uma função delta de Dirac suavizada, definida como

$$\delta(\phi) = \begin{cases} 0, & \text{se } \phi < -\epsilon \\ \frac{1}{2\epsilon} [1 + \cos(\frac{\pi\phi}{\epsilon})], & \text{se } |\phi| \leq \epsilon \\ 0, & \text{se } \phi > \epsilon, \end{cases} \quad (36)$$

em que

$$\epsilon = \frac{3}{2} \max(\Delta x, \Delta y, \Delta z). \quad (37)$$

Com esta definição da função delta de Dirac, nota-se que a interface é a região definida por  $-\epsilon \leq \phi \leq \epsilon$ , de modo que a largura da interface é definida como  $2\epsilon/|\nabla\phi|$ . Deste modo, grandes variações de  $|\nabla\phi|$  acarretam em grandes variações na espessura da interface, com valores muito grandes de  $|\nabla\phi|$  reduzindo consideravelmente a espessura da interface e fazendo com que ela não seja capturada adequadamente, por exemplo. Deste modo, grandes variações de  $|\nabla\phi|$  reduzem a estabilidade do método, e impedem a simulação de escoamentos em que as tensões interfaciais sejam grandes.

Além das características do método de Level Set já descritas, também é útil definir outras propriedades, que serão úteis para fins de pós-processamento dos resultados numéricos obtidos. Da mesma forma como feito para a função delta de Dirac, é interessante definir uma forma suavizada para a função Heaviside. Isto é feito da forma:

$$H(\phi) = \begin{cases} 0, & \text{se } \phi < -\epsilon \\ \frac{1}{2}[1 + \frac{\phi}{\epsilon} + \frac{1}{\pi} \sin(\frac{\pi\phi}{\epsilon})], & \text{se } |\phi| \leq \epsilon \\ 1, & \text{se } \phi > \epsilon, \end{cases} \quad (38)$$

em que o valor de  $\epsilon$  utilizado é o mesmo que para a função delta de Dirac. Com isto, podemos definir integrais de volume no interior da gota. Para uma função genérica  $F$ , temos que sua integral no interior da gota é facilmente definida com a utilização da função Heaviside, na forma:

$$\int_{V_g} F dV = \int_V F H(-\phi) dV, \quad (39)$$

em que  $V_g$  é o volume definido pelo interior da gota, e  $V$  é o volume total do domínio. Para se calcular o volume da gota, por exemplo, utiliza-se  $F = 1$ . Para o caso de integrais de superfície da gota, estas podem ser facilmente definidas com a utilização da função delta de Dirac e da Eq. 32, na forma:

$$\int_{\Gamma} F dS = \int_V F \delta(\phi) |\nabla\phi| dV, \quad (40)$$

em que  $\Gamma$  é a área definida como a superfície da gota. Nota-se que isto é de grande utilidade, tendo-se em vista que o cálculo de integrais de superfície sobre a gota, sem a utilização desta identidade, seria extremamente complexo.

## 2.4 O ALGORITMO DE REINICIALIZAÇÃO

Conforme discutido na seção anterior, é essencial que sejam empregados mecanismos que garantam que a função Level Set  $\phi$  permaneça suficientemente próxima de uma função distância durante toda a simulação. Um destes métodos, que é o utilizado neste

trabalho, se baseia na solução de uma função de reinicialização, originalmente apresentada por Sussman, Smereka e Osher (1994), e se mostra relativamente rápido e de fácil implementação. Esta função de reinicialização é definida como

$$\frac{\partial \phi}{\partial \tau} = S(\phi)(1 - |\nabla \phi|), \quad (41)$$

em que  $S(\phi)$  é uma função sinal, definida como:

$$S(\phi) = \begin{cases} -1, & \text{se } \phi < 0 \\ 0, & \text{se } \phi = 0 \\ 1, & \text{se } \phi > 0. \end{cases} \quad (42)$$

Nota-se que a solução em regime permanente desta equação resulta em  $|\nabla \phi| = 1$ , e que esta função não altera os valores da função Level Set nos pontos em que  $\phi = 0$ , preservando a posição da interface durante o processo de reinicialização. Deste modo, o algoritmo de reinicialização consiste em evoluir a Eq. 41 até um instante  $\tau$  em que ela esteja suficientemente próxima do regime permanente. Vale notar que a Eq. 41 pode ser reescrita como uma equação de advecção,

$$\frac{\partial \phi}{\partial \tau} = -\mathbf{w} \cdot \nabla \phi + S(\phi), \quad (43)$$

em que

$$\mathbf{w} = S(\phi) \frac{\nabla \phi}{|\nabla \phi|}. \quad (44)$$

Com a equação reescrita nesta forma, fica evidente que o algoritmo de reinicialização consiste, essencialmente, em um transporte de informação a partir da interface, para fora. Deste modo, a solução desta equação converge primeiro nas regiões próximas à interface - que são as regiões em que a reinicialização do Level Set realmente é importante. Além disso, também é evidente que a utilização de esquemas de *upwinding* é essencial no algoritmo de reinicialização e, como o campo de velocidades  $\mathbf{w}$  não é previamente definido, sendo dependente dos valores de  $\phi$ , a implementação destes esquemas não é trivial, especialmente sobre a interface, em que  $\mathbf{w}$  é descontínuo. Esta questão será abordada com mais detalhes na seção sobre discretização espacial da equação de reinicialização.

Para melhorar o desempenho e reduzir erros associados à solução numérica desta equação, é interessante utilizar uma função sinal suavizada,  $S_\epsilon$ . Uma possível definição para esta função sinal é a utilizada por Sussman, Smereka e Osher (1994):

$$S_\epsilon = \frac{\phi_0}{\sqrt{\phi_0^2 + \epsilon^2}}, \quad (45)$$

em que  $\phi_0$  é a função Level Set antes do início do algoritmo de reinicialização, e  $\epsilon$  é um parâmetro de suavização, que também evita a ocorrência de singularidades. Tipicamente,

são utilizadas malhas quadradas, em que  $\Delta x = \Delta y = \Delta z$ , e este parâmetro de suavização é definido como  $\epsilon = \Delta x$ . Nota-se que, com esta definição, a função sinal só precisa ser calculada uma vez, no início do processo de reinicialização.

Neste trabalho, entretanto, será utilizada uma outra definição para a função sinal:

$$S_\epsilon = \frac{\phi}{\sqrt{\phi^2 + |\nabla\phi|^2\epsilon^2}}, \quad (46)$$

em que  $\epsilon = \max(\Delta x, \Delta y, \Delta z)$ . Esta definição, proposta por Peng et al. (1999), altera o valor da função sinal com base nos valores de  $\nabla\phi$ , de modo que a convergência da função de reinicialização é acelerada nos casos em que  $|\nabla\phi| < 1$ , reduzindo o número de iterações necessárias no processo de reinicialização, ao passo que a convergência é desacelerada nos casos em que  $|\nabla\phi| > 1$ , aumentando a estabilidade do processo de reinicialização e evitando possíveis mudanças de posição da interface, ocasionadas por erros numéricos. Nota-se que, com esta definição, a função sinal deve ser continuamente atualizada durante o processo de reinicialização.

## 2.5 O MÉTODO DE LEVEL SET LOCAL

Como evidenciado pela presença do delta de Dirac na Eq. 34, a função Level Set só é utilizada em uma estreita região nas proximidades da interface, de forma que quaisquer operações realizadas sobre a função Level Set em pontos distantes da interface são, essencialmente, inúteis e um desperdício de recursos computacionais. Deste modo, a implementação de um método de Level Set local se mostra extremamente vantajosa, isto é, a implementação de um método em que a função Level Set seja calculada somente na região próxima à interface.

Neste trabalho, é utilizada a metodologia apresentada por Peng et al. (1999), que consiste na criação de duas regiões nas proximidades da interface, e na introdução de uma função *cutoff* ( $C(\phi)$ ) na equação de transporte do Level Set. A primeira região, mais interna, é definida como os pontos nos quais  $|\phi| \leq \beta$ , em que  $\beta = 3 \max(\Delta x, \Delta y, \Delta z)$ , enquanto a segunda região, mais abrangente, é definida como os pontos em que  $|\phi| \leq \gamma$ , em que  $\gamma = 6 \max(\Delta x, \Delta y, \Delta z)$ .

Com a introdução da função *cutoff*, a Eq. 27, de transporte de  $\phi$ , é modificada para

$$\frac{\partial\phi}{\partial t} = -C(\phi)\mathbf{u} \cdot \nabla\phi, \quad (47)$$



em que a função *cutoff* é definida como

$$C(\phi) = \begin{cases} 1, & \text{se } |\phi| \leq \beta \\ (|\phi| - \gamma)^2(2|\phi| + \gamma - 3\beta)/(\gamma - \beta)^3, & \text{se } \beta < |\phi| \leq \gamma \\ 0, & \text{se } |\phi| > \gamma. \end{cases} \quad (48)$$

Deste modo, a região mais interna ( $|\phi| \leq \beta$ ) é a região na qual a definição da função Level Set é mais importante, enquanto a região mais externa ( $\beta < |\phi| \leq \gamma$ ) atua como uma zona de transição entre a região mais interna e o restante do domínio, onde a função Level Set não é calculada. Como o processo de reinicialização consiste basicamente em uma propagação de informação a partir da interface e em direção normal, das regiões mais próximas à interface para as regiões mais externas, a utilização de uma função *cutoff* não se faz necessária, pois informações de regiões mais externas não afetam as regiões mais internas. Deste modo, as operações são realizadas na região mais abrangente ( $|\phi| \leq \gamma$ ), sem a utilização de uma função *cutoff*. Por fim, a função Level Set é definida no resto do domínio como

$$\phi = \begin{cases} -\gamma, & \text{se } \phi < -\gamma \\ \phi, & \text{se } |\phi| \leq \gamma \\ \gamma, & \text{se } \phi > \gamma. \end{cases} \quad (49)$$

## 2.6 DISCRETIZAÇÃO TEMPORAL DAS EQUAÇÕES DE TRANSPORTE E REINICIALIZAÇÃO DO LEVEL SET

Tanto a equação de transporte do Level Set quanto a equação de reinicialização são equações de advecção e, portanto, possuem condições de estabilidade fortemente dependentes da utilização de esquemas de *upwinding*, e pouco dependentes da utilização de métodos implícitos. Deste modo, estas equações serão evoluídas no tempo com a utilização de métodos explícitos, simplificando a implementação de esquemas de *upwinding* para a discretização espacial, bem como reduzindo o custo computacional, tendo em vista que métodos explícitos não requerem a solução de sistemas lineares.

Para a evolução no tempo da equação de transporte do Level Set e da equação de reinicialização, temos essencialmente um problema de valor inicial, de forma que o método padrão de solução é a utilização de esquemas de Runge-Kutta. Além disso, para garantir a estabilidade do método numérico, também deseja-se que o esquema utilizado seja TVD. Deste modo, o esquema de discretização temporal escolhido é um Runge-Kutta TVD de terceira ordem, descrito por Shu e Osher (1988). Apesar de existirem métodos Runge-Kutta TVD de ordens maiores, estes requerem a utilização de esquemas de *downwinding*, em adição aos esquemas de *upwinding*, de modo que a complexidade e o

custo computacional se tornam muito maiores, e a sua utilização não se mostra vantajosa para esta aplicação.

Para uma equação genérica

$$\frac{\partial \phi}{\partial t} = L(\phi, t), \quad (50)$$

em que  $L(\phi, t)$  é um operador genérico, o método Runge-Kutta TVD de terceira ordem é dado por:

$$\begin{cases} \phi_1 = \phi_0 + \Delta t L(\phi_0, t_0) \\ \phi_2 = \frac{3}{4}\phi_0 + \frac{1}{4}\phi_1 + \frac{1}{4}\Delta t L(\phi_1, t_1) \\ \phi_3 = \frac{1}{3}\phi_0 + \frac{2}{3}\phi_2 + \frac{2}{3}\Delta t L(\phi_2, t_2), \end{cases} \quad (51)$$

em que  $\phi_0$  é o valor de  $\phi$ , já definido, num instante de tempo  $t^n$ , e  $\phi_3$  é o valor de  $\phi$  calculado no instante  $t^{n+1}$ . Alternativamente, este mesmo método também pode ser descrito como uma combinação linear de aplicações do método de Euler explícito, na forma:

$$\begin{cases} \tilde{\phi}^{n+1} = \phi^n + \Delta t L(\phi^n, t^n) \\ \tilde{\phi}^{n+2} = \tilde{\phi}^{n+1} + \Delta t L(\tilde{\phi}^{n+1}, t^n + \Delta t) \\ \tilde{\phi}^{n+\frac{1}{2}} = \frac{3}{4}\phi^n + \frac{1}{4}\tilde{\phi}^{n+2} \\ \tilde{\phi}^{n+\frac{3}{2}} = \tilde{\phi}^{n+\frac{1}{2}} + \Delta t L(\tilde{\phi}^{n+\frac{1}{2}}, t + \frac{1}{2}\Delta t) \\ \phi^{n+1} = \frac{1}{3}\phi^n + \frac{2}{3}\tilde{\phi}^{n+\frac{3}{2}}. \end{cases} \quad (52)$$

Para o caso da Eq. 27, de transporte do Level Set, temos que

$$L(\phi^n, t^n) = -\mathbf{u}(t^n) \cdot \nabla \phi^n. \quad (53)$$

Como, no código numérico, a evolução do Level Set é realizada após a solução das equações de Navier-Stokes, os valores de  $\mathbf{u}^n$  e  $\mathbf{u}^{n+1}$  já são conhecidos, e os valores de  $\mathbf{u}^{n+\frac{1}{2}}$  são facilmente obtidos por interpolação. Já para o caso da Eq. 41, de reinicialização do Level Set, o operador  $L$  é definido como

$$L(\phi^n) = S_\epsilon(\phi^n)(1 - |\nabla \phi^n|), \quad (54)$$

e depende unicamente dos valores de  $\phi$ . Para o caso do algoritmo de reinicialização, as iterações são realizadas em um domínio de tempo auxiliar,  $\tau$ , de modo que tanto os passos de tempo ( $\Delta\tau$ ) quanto o número de iterações realizadas (valor final de  $\tau$ ) são independentes das iterações no domínio de tempo principal ( $t$ ). Além disso, não é necessário que o algoritmo de reinicialização seja utilizado em todas as iterações do programa principal. Deste modo, é necessário estabelecer valores para  $\Delta\tau$ , bem como número de iterações do algoritmo de reinicialização e a frequência de utilização deste algoritmo.

O número de iterações e a frequência de reinicializações não são fixos, e serão alterados em função da necessidade. escoamentos em que a função Level Set sofre poucas variações - cisalhamento simples com baixos números de capilaridade, por exemplo - requerem uma menor frequência de reinicializações do que escoamentos com grandes variações da função Level Set. Como erros numéricos no algoritmo de reinicialização podem ocasionar mudanças na posição da interface, gerando uma perda de volume da gota, é interessante que o número de operações de reinicialização seja mantido o mínimo possível, desde que suficiente para manter a função Level Set suficientemente próxima de uma função distância. Os passos de tempo  $\Delta\tau$  são definidos como:

$$\Delta\tau = \frac{1}{4} \min(\Delta x, \Delta y, \Delta z). \quad (55)$$

Analisando a Eq. 43, nota-se que a velocidade característica  $\mathbf{w}$  sempre possuirá norma próxima de 1. Deste modo, no pior dos casos, o número de  $CFL$  é

$$CFL = \frac{|\mathbf{w}|\Delta\tau}{\min(\Delta x, \Delta y, \Delta z)} \approx \frac{1}{4}, \quad (56)$$

garantindo a estabilidade do algoritmo de reinicialização.

## 2.7 DISCRETIZAÇÃO EM DIFERENÇAS FINITAS

### 2.7.1 Aspectos gerais

Neste trabalho, utiliza-se uma malha escalonada, semelhante à utilizada por Kim e Moin (1985), porém expandida para o caso tridimensional, de modo que os nós para as propriedades do escoamento não são situados nas mesmas posições. Para isto, divide-se o domínio em cubóides, nos quais os nós para grandezas escalares, como  $P, \chi$  e  $\phi$ , se situam no centro volumétrico destes elementos, e os nós para grandezas vetoriais, como  $u, v, w, u^*, v^*$  e  $w^*$ , se situam nos centros das faces destes elementos. Esta disposição pode ser visualizada na Fig. 3, que representa um dos cubóides da malha, bem como a posição dos nós em cada um destes cubóides. Parte do elemento foi removida, de modo a permitir a visualização do nó de  $P$  em seu centro.

Para ajudar a exemplificar o conceito de malha escalonada, a Fig. 4 apresenta uma esquematização de malha escalonada bidimensional, onde é possível observar as posições dos nós de velocidade nas faces, e dos nós de pressão no centro dos elementos.

Com a discretização do problema por meio do método das diferenças finitas, deseja-se reescrever as Eqs. 10 e 12 como sistemas lineares descritos na forma de estêncil, de acordo com a Eq. 57:

$$a_{ijk}x_{i-1jk} + b_{ijk}x_{ij-1k} + c_{ijk}x_{ijk-1} + d_{ijk}x_{ijk} + e_{ijk}x_{i+1jk} + f_{ijk}x_{ij+1k} + g_{ijk}x_{ijk+1} = h_{ijk}, \quad (57)$$

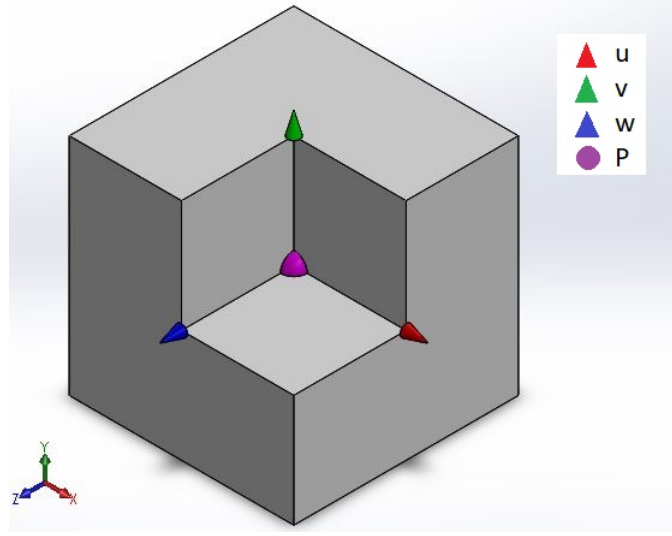


Figura 3 – Esquematização de um elemento da malha escalonada tridimensional

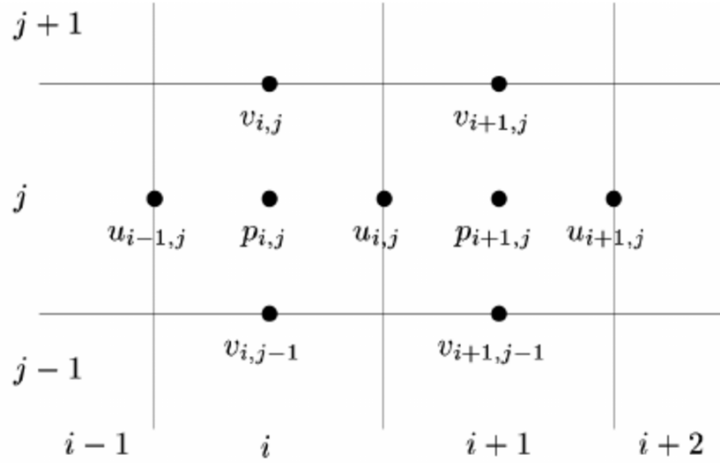


Figura 4 – Esquematização de uma malha escalonada bidimensional

conforme apresentado em Ferziger e Peric (2012). Deste modo, obtêm-se quatro sistemas lineares distintos, para cada intervalo de tempo, sendo eles para  $u^*$ ,  $v^*$ ,  $w^*$  e  $\chi$ . Estes sistemas lineares são então resolvidos com a utilização do método dos Gradientes Conjugados, com um preconditionador do tipo SSOR. Este método é descrito em detalhes em seções subsequentes.

As derivadas são discretizadas, exceto quando explicitado, de acordo com o método de diferenças finitas apresentado em Saad (2003), de forma centrada, e são aproximações de segunda ordem. Para uma função genérica  $\psi$ , as aproximações para as derivadas primeira e segunda são, respectivamente,

$$\left. \frac{\partial \psi}{\partial x} \right|_{ijk} = \frac{\psi_{i+1jk} - \psi_{i-1jk}}{2\Delta x}, \quad (58)$$

$$\left. \frac{\partial^2 \psi}{\partial x^2} \right|_{ijk} = \frac{\psi_{i+1jk} - 2\psi_{ijk} + \psi_{i-1jk}}{\Delta x^2}. \quad (59)$$

As derivadas nas outras direções são discretizadas de forma análoga. Deve-se atentar ao fato de que, em função da utilização de uma malha escalonada, as propriedades do escoamento, incluindo suas derivadas, são localizadas em pontos diferentes. Deste modo, em alguns casos é desejado que a uma derivada seja calculada em um ponto situado entre dois nós da propriedade a ser derivada. Para isso, utiliza-se a discretização a seguir:

$$\left. \frac{\partial \psi}{\partial x} \right|_{i+\frac{1}{2}jk} = \frac{\psi_{i+1jk} - \psi_{ijk}}{\Delta x}, \quad (60)$$

em que a derivada é centrada, tendo em vista que ela está definida em um ponto situado entre os nós  $(i, j, k)$  e  $(i + 1, j, k)$ .

A seguir são apresentadas como exemplo as discretizações das principais equações governantes do problema estudado, bem como a aplicação de suas condições de contorno. A Tab. 9 apresenta os parâmetros utilizados nos sistemas lineares na forma de estêncil. No desenvolvimento do método de solução das equações de Navier-Stokes são utilizadas somente derivadas centradas ou esquemas ENO de segunda ordem, de forma que o método como um todo também seja de segunda ordem. Para a discretização das equações de evolução e transporte do Level Set, são utilizados esquemas de maior ordem.

### 2.7.2 Esquemas ENO e WENO

Para o cálculo das derivadas utilizadas em esquemas de *upwinding*, tanto para o termo convectivo das equações de Navier-Stokes quanto para as equações de transporte e reinicialização do Level Set, a utilização de esquemas simples de *forward differencing* e *backwards differencing* com estêncil de dois pontos não é adequada, tendo em vista que estes esquemas são de primeira ordem apenas. Deste modo, se faz necessária a utilização de esquemas de ordem maior e, portanto, mais complexos. Para isso, são utilizados esquemas ENO (*Essentially Non-Oscillatory*) e WENO (*Weighted Essentially Non-Oscillatory*).

O esquema ENO consiste em, basicamente, utilizar polinômios interpoladores e então calcular as derivadas com base nestes polinômios. Como indicado pelo nome, este esquema assume que a função a ser derivada não possui grandes oscilações e, deste modo, utiliza a alternativa mais suave dentre as opções de polinômios interpoladores. Já o esquema WENO consiste em realizar uma média ponderada de todos os valores possíveis de serem obtidos por um esquema ENO de mesmo estêncil, ao invés de apenas o relativo ao polinômio mais suave, de forma que é possível se obter aproximações de maior ordem para as derivadas. Para os casos em que os dados fornecidos forem suaves, o esquema WENO utiliza pesos para a média ponderada que resultem na maior ordem possível para a aproximação, ao passo em que reduz os pesos de aproximações que se mostrem pouco suaves, em detrimento da ordem de aproximação, de modo a preservar a suavidade do resultado. Em casos extremos, o esquema WENO converge para resultados iguais aos de um esquema ENO. Nesta seção são apresentadas as definições destes esquemas para um

caso genérico unidimensional, tendo em vista que a aplicação deste método no caso tridimensional consiste unicamente em aplicar o esquema unidimensional ao longo de cada uma das dimensões desejadas.

Para o cálculo das derivadas utilizadas no termo convectivo das equações de Navier-Stokes, é utilizado um esquema ENO de segunda ordem. Como todos os outros termos da equação também são aproximações de segunda ordem, a utilização de derivadas de maior ordem neste termo é desnecessária. O esquema aqui apresentado é o descrito por Osher e Fedkiw (2006).

Com a utilização de uma variável auxiliar  $k$ , definida como  $k = i - 1$ , caso se deseje calcular a derivada para trás  $\phi_x^-$ , ou  $k = i$  caso de deseje calcular a derivada para frente  $\phi_x^+$ . Define-se então:

$$\begin{cases} D_{k+\frac{1}{2}}^1 = \frac{\phi_{k+1} - \phi_k}{\Delta x} \\ D_k^2 = \frac{\phi_{k+1} - 2\phi_k + \phi_{k-1}}{2\Delta x^2} \\ D_{k+1}^2 = \frac{\phi_{k+2} - 2\phi_{k+1} + \phi_k}{2\Delta x^2} \end{cases} \quad (61)$$

Para escolher qual das aproximações utilizar, o esquema ENO decide sempre pela alternativa mais suave, isto é, a que apresentar menor derivada. Deste modo, define-se uma segunda variável auxiliar  $c$  como  $c = D_k^2$  para o caso em que  $|D_k^2| \leq |D_{k+1}^2|$ , ou como  $c = D_{k+1}^2$  para o caso em que  $|D_k^2| > |D_{k+1}^2|$ . Com isto, uma aproximação de segunda ordem para a derivada pode ser obtida, na forma:

$$\begin{cases} Q'_1 = D_{k+\frac{1}{2}}^1 \\ Q'_2 = c(2(i - k) - 1)\Delta x \\ \phi_x = Q'_1 + Q'_2 \end{cases} \quad (62)$$

Já para o caso das equações de transporte e reinicialização da função Level Set, deseja-se obter a maior precisão espacial possível, tendo em vista que erros numéricos causam uma perda de massa da gota. Deste modo, nestas equações é utilizado um esquema WENO de quinta ordem, desenvolvido por Jiang e Shu (1996), para o cálculo das derivadas. O esquema aqui apresentado, entretanto, se encontra na forma como foi descrito por Osher e Fedkiw (2006), para uma maior clareza e facilidade de aplicação.

Para o cálculo de uma derivada para trás,  $\phi_x^-$ , inicialmente define-se os parâmetros:

$$\begin{cases} v_1 = \frac{\phi_{i-2} - \phi_{i-3}}{\Delta x} \\ v_2 = \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x} \\ v_3 = \frac{\phi_i - \phi_{i-1}}{\Delta x} \\ v_4 = \frac{\phi_{i+1} - \phi_i}{\Delta x} \\ v_5 = \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x} \end{cases} \quad (63)$$

Alternativamente, caso se deseje calcular a derivada para frente ( $\phi_x^+$ ), estes parâmetros são definidos como:

$$\begin{cases} v_1 = \frac{\phi_{i+3} - \phi_{i+2}}{\Delta x} \\ v_2 = \frac{\phi_{i+2} - \phi_{i+1}}{\Delta x} \\ v_3 = \frac{\phi_{i+1} - \phi_i}{\Delta x} \\ v_4 = \frac{\phi_i - \phi_{i-1}}{\Delta x} \\ v_5 = \frac{\phi_{i-1} - \phi_{i-2}}{\Delta x} . \end{cases} \quad (64)$$

Com estes parâmetros definidos, o restante dos cálculos é idêntico para as duas derivadas. Para tal, inicialmente define-se as três aproximações possíveis para a derivada, obtidas por um esquema ENO de mesmo estêncil. São elas:

$$\begin{cases} \phi_x^1 = \frac{v_1}{3} - \frac{7v_2}{6} + \frac{11v_3}{6} \\ \phi_x^2 = -\frac{v_2}{6} + \frac{5v_3}{6} + \frac{v_4}{3} \\ \phi_x^3 = \frac{v_3}{3} + \frac{5v_4}{6} - \frac{v_5}{6} . \end{cases} \quad (65)$$

A estimativa para a derivada obtida pelo esquema WENO consiste em uma média ponderada destas três aproximações. Os valores ótimos para os pesos, que resultam em uma aproximação de quinta ordem, são, respectivamente, 0, 1, 0, 6 e 0, 3. Entretanto, caso algum dos estêncis não seja suave, isto pode gerar erros consideráveis. Deste modo, é desejável estabelecer um esquema que reduza os pesos de estêncis pouco suaves, de forma a eliminar este erro, mesmo em detrimento da ordem da aproximação.

Para calcular estes pesos variáveis, o primeiro passo é definir a suavidade dos estêncis associados a cada uma das aproximações:

$$\begin{cases} S_1 = \frac{13}{12}(v_1 - 2v_2 + v_3)^2 + \frac{1}{4}(v_1 - 4v_2 + 3v_3)^2 \\ S_2 = \frac{13}{12}(v_2 - 2v_3 + v_4)^2 + \frac{1}{4}(v_2 - v_4)^2 \\ S_3 = \frac{13}{12}(v_3 - 2v_4 + v_5)^2 + \frac{1}{4}(3v_3 - 4v_4 + v_5)^2 , \end{cases} \quad (66)$$

que são utilizadas para calcular os parâmetros:

$$\begin{cases} \alpha_1 = \frac{0,1}{(S_1 + \epsilon)^2} \\ \alpha_2 = \frac{0,6}{(S_2 + \epsilon)^2} \\ \alpha_3 = \frac{0,3}{(S_3 + \epsilon)^2} \\ \epsilon = 10^{-6} . \end{cases} \quad (67)$$

Estes parâmetros, quando normalizados, correspondem os pesos a serem utilizados no cálculo da derivada. Nota-se, que para os casos em que os valores de suavidade são próximos, estes pesos se aproximam dos valores ótimos para obter uma derivada de quinta

ordem. A normalização destes parâmetros é feita na forma:

$$\begin{cases} \omega_1 = \frac{\alpha_1}{\alpha_1 + \alpha_2 + \alpha_3} \\ \omega_2 = \frac{\alpha_2}{\alpha_1 + \alpha_2 + \alpha_3} \\ \omega_3 = \frac{\alpha_3}{\alpha_1 + \alpha_2 + \alpha_3} \end{cases} \quad (68)$$

Por fim, a aproximação para a derivada, calculada pelo esquema WENO, é dada por

$$\phi_x = \omega_1 \phi_x^1 + \omega_2 \phi_x^2 + \omega_3 \phi_x^3. \quad (69)$$

Nota-se que o esquema ENO de segunda ordem aqui apresentado pode originar um esquema WENO de terceira ordem, ao passo que o esquema WENO de quinta ordem se origina de um esquema ENO de terceira ordem. Entretanto, como estes esquemas não são utilizados neste trabalho, eles não serão apresentados em maiores detalhes.

### 2.7.3 Equação da velocidade-tentativa

Nesta seção, será apresentada a discretização das equações evolutivas de  $u$ , como exemplo. As discretizações das equações de  $v$  e  $w$  são feitas de forma análoga. Primeiramente, a componente em  $x$  da Eq. 34 é reescrita como

$$u^* - \frac{1}{2Re} \nabla^2 u^* = u^n - \Delta t [\mathbf{u} \cdot \nabla u]^{n+\frac{1}{2}} + \frac{\Delta t}{2Re} \nabla^2 u^n - \frac{\Delta t}{ReCa} \left[ \kappa \delta(\phi) \frac{\partial \phi}{\partial x} \right]^{n+\frac{1}{2}}. \quad (70)$$

Nota-se que os termos do lado direito da Eq. 70 não possuem  $u^*$ , e portanto correspondem ao parâmetro  $h$  do sistema linear discretizado na forma do estêncil, enquanto os termos do lado esquerdo correspondem aos demais parâmetros. Também vale notar que, como o termo convectivo  $\mathbf{u} \cdot \nabla \mathbf{u}$  é calculado de forma explícita, a implementação de um esquema de *upwinding* é trivial, e contribui para uma maior estabilidade do método. De forma a manter a solução das equações de Navier-Stokes de segunda ordem, e evitar um aumento de complexidade e custo computacional desnecessários, este termo é calculado com a utilização de um esquema ENO de segunda ordem. A discretização deste termo, bem como a discretização do termo relativo às tensões viscosas, são realizadas como:

$$\nabla^2 u = \frac{u_{i-1jk} - 2u_{ijk} + u_{i+1jk}}{\Delta x^2} + \frac{u_{ij-1k} - 2u_{ijk} + u_{ij+1k}}{\Delta y^2} + \frac{u_{ijk-1} - 2u_{ijk} + u_{ijk+1}}{\Delta z^2}, \quad (71)$$

$$\mathbf{u} \cdot \nabla u = u_{ijk} \frac{\partial u}{\partial x} + \bar{v} \frac{\partial u}{\partial y} + \bar{w} \frac{\partial u}{\partial z}, \quad (72)$$

$$\bar{v} = \frac{v_{ijk} + v_{i+1jk} + v_{ij-1k} + v_{i+1j-1k}}{4}, \quad (73)$$

$$\bar{w} = \frac{w_{ijk} + w_{i+1jk} + w_{ijk-1} + w_{i+1jk-1}}{4}, \quad (74)$$



em que  $\bar{v}$  e  $\bar{w}$  são velocidades calculadas, por interpolação, na posição dos nós de  $u$ . Isto se faz necessário devido à utilização de uma malha escalonada. Como o esquema utilizado no termo convectivo é de *upwinding*, caso  $u > 0$  utiliza-se  $u_x = u_x^-$ , e caso  $u < 0$  utiliza-se  $u_x = u_x^+$ , com os termos relativos às outras duas componentes da velocidade calculados de forma análoga.

A componente relativa a  $\nabla\phi$  é dada por

$$\frac{\partial\phi}{\partial x} = \frac{\phi_{i+1jk} - \phi_{ijk}}{\Delta x}. \quad (75)$$

É importante observar que os valores de  $\phi$  são localizados no centro dos elementos de malha, enquanto os valores das velocidades são localizados nas faces. Deste modo, como a Eq. 70 se refere a pontos localizados nas faces, todos os seus termos também devem ser calculados nas faces, e a discretização apresentada na Eq. 75 é uma derivada centrada.

Por fim, os termos relativos à curvatura são discretizados como:

$$\kappa = \frac{1}{2}\kappa_{i+1jk} + \frac{1}{2}\kappa_{ijk}, \quad (76)$$

$$\kappa_{ijk} = \frac{n_{x_{i+\frac{1}{2}jk}} - n_{x_{i-\frac{1}{2}jk}}}{\Delta x} + \frac{n_{y_{ij+\frac{1}{2}k}} - n_{y_{ij-\frac{1}{2}k}}}{\Delta y} + \frac{n_{z_{ijk+\frac{1}{2}}} - n_{z_{ijk-\frac{1}{2}}}}{\Delta z}, \quad (77)$$

$$n_{x_{i+\frac{1}{2}jk}} = \frac{\phi_{i+1jk} - \phi_{ijk}}{\Delta x}, \quad (78)$$

$$n_{y_{ij+\frac{1}{2}k}} = \frac{\phi_{ij+1k} - \phi_{ijk}}{\Delta y}, \quad (79)$$

$$n_{z_{ijk+\frac{1}{2}}} = \frac{\phi_{ijk+1} - \phi_{ijk}}{\Delta x}. \quad (80)$$

Nesta discretização, nota-se que as componentes da normal são definidas nas faces dos elementos de malha, ao passo em que as derivadas destas componentes e, portanto, o termo de curvatura, são definidos nos centros destes elementos de malha. Deste modo, o termo relativo à curvatura utilizado na Eq. 70 deve ser estimado por interpolação, de modo que ele esteja definido sobre a face do elemento de malha, nos mesmos pontos que os nós de  $u$ .

#### 2.7.4 Equação da pressão

Para montar o sistema linear na forma de estêncil correspondente à pressão, discretiza-se a Eq. 12:

$$\left. \frac{\partial^2 \chi}{\partial x^2} \right|_{ijk} + \left. \frac{\partial^2 \chi}{\partial y^2} \right|_{ijk} + \left. \frac{\partial^2 \chi}{\partial z^2} \right|_{ijk} = \frac{1}{\Delta t} [\nabla \cdot \mathbf{u}^*]_{ijk}, \quad (81)$$

$$\left. \frac{\partial^2 \chi}{\partial x^2} \right|_{ijk} = \frac{\chi_{i-1jk}^{n+1} - 2\chi_{ijk}^{n+1} + \chi_{i+1jk}^{n+1}}{\Delta x^2}, \quad (82)$$

$$\left. \frac{\partial^2 \chi}{\partial y^2} \right|_{ijk} = \frac{\chi_{ij-1k}^{n+1} - 2\chi_{ijk}^{n+1} + \chi_{ij+1k}^{n+1}}{\Delta y^2}, \quad (83)$$

$$\left. \frac{\partial^2 \chi}{\partial z^2} \right|_{ijk} = \frac{\chi_{ijk-1}^{n+1} - 2\chi_{ijk}^{n+1} + \chi_{ijk+1}^{n+1}}{\Delta z^2}, \quad (84)$$

$$[\nabla \cdot \mathbf{u}^*]_{ijk} = \frac{u_{ijk}^* - u_{i-1jk}^*}{\Delta x} + \frac{u_{ijk}^* - u_{ij-1k}^*}{\Delta y} + \frac{u_{ijk}^* - u_{ijk-1}^*}{\Delta z}. \quad (85)$$

Nota-se que o termo do lado direito da Eq. 81 corresponde ao parâmetro  $h$  do sistema linear discretizado na forma de estêncil, enquanto os termos do lado esquerdo correspondem aos outros parâmetros. Vale notar que as derivadas de  $\mathbf{u}^*$  são centradas e, portanto, de segunda ordem. Isto ocorre pois estas derivadas são computadas em nós no centro de cada bloco da malha (posição dos nós de  $\chi$ ), enquanto os nós das velocidades se encontram nas faces de cada bloco.

### 2.7.5 Equação da velocidade real

Com base nos valores calculados para  $\mathbf{u}^*$  e  $\chi^{n+1}$ , é possível determinar a velocidade real do escoamento,  $\mathbf{u}^{n+1}$ , utilizando-se a Eq. 13. Esta equação é discretizada como

$$u_{ijk}^{n+1} = u_{ijk}^* - \Delta t \frac{\chi_{i+1jk}^{n+1} - \chi_{ijk}^{n+1}}{\Delta x}, \quad (86)$$

$$v_{ijk}^{n+1} = v_{ijk}^* - \Delta t \frac{\chi_{ij+1k}^{n+1} - \chi_{ijk}^{n+1}}{\Delta y}, \quad (87)$$

$$w_{ijk}^{n+1} = w_{ijk}^* - \Delta t \frac{\chi_{ijk+1}^{n+1} - \chi_{ijk}^{n+1}}{\Delta z}. \quad (88)$$

Vale notar que as derivadas de  $\chi$  são computadas nas faces de cada um dos blocos da malha, onde estão situados os nós das velocidades, e, portanto, são centradas e de segunda ordem.

### 2.7.6 Equações de transporte e reinicialização do Level Set

Para a discretização das equações de transporte e reinicialização do Level Set, a metodologia utilizada é diferente, tendo em vista que suas derivadas são calculadas com esquemas de *upwinding* de alta ordem, de modo a garantir sua estabilidade e a minimizar ao máximo os erros numéricos, que ocasionam uma perda de volume da gota. Deste modo, em ambas as equações são utilizados os esquemas WENO de quinta ordem. Como descrito

na seção 2.6, o método Runge-Kutta TVD utilizado pode ser descrito como uma série de aplicações do método de Euler explícito. Deste, modo, nesta seção serão apresentadas apenas a discretização de uma destas aplicações para cada caso, tendo em vista que as outras duas aplicações são feitas de forma análoga.

Para o caso da equação do transporte, a Eq. 27 pode ser discretizada como:

$$\phi_{ijk}^{n+1} = \phi_{ijk}^n - \Delta t \left( \bar{u} \frac{\partial \phi}{\partial x} + \bar{v} \frac{\partial \phi}{\partial y} + \bar{w} \frac{\partial \phi}{\partial z} \right), \quad (89)$$

em que  $\bar{u}, \bar{v}$  e  $\bar{w}$  são os valores das velocidades calculados, por interpolação, nos centros dos elementos de malha, onde estão situados os nós de  $\phi$ . Estes valores são calculados como:

$$\bar{u} = \frac{u_{ijk}^n + u_{i-1jk}^n}{2}, \quad (90)$$

$$\bar{v} = \frac{v_{ijk}^n + v_{ij-1k}^n}{2}, \quad (91)$$

$$\bar{w} = \frac{w_{ijk}^n + w_{ijk-1}^n}{2}. \quad (92)$$

As derivadas de  $\phi$  são calculadas com a utilização do esquema WENO de quinta ordem, com  $\phi_x = \phi_x^-$  para o caso em que  $\bar{u} > 0$ , e  $\phi_x = \phi_x^+$  para o caso em que  $\bar{u} < 0$ , de acordo com o esquema de *upwinding*. Os termos relativos às outras duas direções são tratados de forma análoga.

A inicialização da função Level Set é feita de forma trivial, já que a gota possui um formato inicialmente esférico. Os valores iniciais são determinados como a distância entre um dado nó e a superfície esférica definida como formato inicial da gota. Isto é feito da forma:

$$\phi_{ijk}^0 = \sqrt{\left( \frac{x_{ijk} + x_{i-1jk}}{2} - x_c \right)^2 + \left( \frac{y_{ijk} + y_{ij-1k}}{2} - y_c \right)^2 + \left( \frac{z_{ijk} + z_{ijk-1}}{2} - z_c \right)^2} - r_g, \quad (93)$$

em que  $r_g$  é o raio inicial da gota.

Como na equação de transporte do Level Set o campo de velocidade  $\mathbf{u}$  já é previamente determinado e contínuo, a implementação de *upwinding* é trivial. Para o caso da equação de reinicialização, esta implementação é consideravelmente mais complicada.

Para entender melhor estas complicações, vamos analisar a definição da velocidade característica da equação de reinicialização, de acordo com a Eq. 43, e definida como:

$$\mathbf{w} = S(\phi) \frac{\nabla \phi}{|\nabla \phi|}. \quad (94)$$

Com esta definição da velocidade característica, pode-se observar, primeiramente, que seus valores não são previamente determinados, de forma que eles variam continuamente durante o processo de reinicialização. Além disso, nota-se que esta velocidade

característica é descontínua na interface, tendo norma sempre próxima de 1 e possuindo o mesmo sinal que  $\phi$ , de modo que deve-se tomar grande cuidado na escolha de como proceder nos nós próximos a interface. Isto faz com que a escolha entre se utilizar  $\phi_x^-$  ou  $\phi_x^+$  como uma aproximação para  $\phi_x$  seja relativamente complexa. Os valores de  $\phi_x^-$  e  $\phi_x^+$  são calculados com a utilização do esquema WENO de quinta ordem, e o mesmo é válido para as derivadas nas outras duas direções.

Para se lidar com a descontinuidade, e determinar a forma correta de cálculo das derivadas na região da interface, é utilizado o método de Godunov. Este método, utilizado para discretizar termos do tipo  $(a\nabla\phi/|\nabla\phi|) \cdot \nabla\phi$ , consiste em, para os casos em que  $\phi_x^-$  e  $\phi_x^+$  possuírem mesmo sinal, utilizar *upwinding* tradicional, já que não há discordância. Para os casos em que  $a\phi_x^- < 0$  e  $a\phi_x^+ > 0$ , entende-se que há um fluxo de informação para fora do ponto em questão. Deste modo, não há um fluxo de informação para o ponto em questão, e o método de Godunov explicita isso ao definir  $\phi_x = 0$  neste ponto. Por fim, para os casos em que  $a\phi_x^- > 0$  e  $a\phi_x^+ < 0$ , entende-se que há um fluxo de informação para dentro do ponto em questão. Para solucionar este problema, o método de Godunov escolhe como solução a informação que chegar primeiro, isto é, dentre  $a\phi_x^-$  e  $a\phi_x^+$ , o método escolhe o que possuir o maior valor absoluto. A discretização dos termos  $\phi_y$  e  $\phi_z$  é feita de forma análoga.

Para o caso da equação de reinicialização, temos que  $a = S(\phi)$ , que depende dos valores de  $\phi_x, \phi_y$  e  $\phi_z$ , que ainda não foram determinados. Entretanto, para utilizar o método de Godunov, não é necessário conhecer o valor de  $S(\phi)$ , mas apenas seu sinal, que, claramente, será igual ao de  $\phi$ . Deste modo, o método de Godunov, conforme esquematizado por Osher e Fedkiw (2006), pode ser descrito, com adaptações, como:

$$\frac{\partial\phi^2}{\partial x} = \begin{cases} \max(\max(\phi_x^-, 0)^2, \min(\phi_x^+, 0)^2), & \text{se } \phi > 0 \\ \max(\min(\phi_x^-, 0)^2, \max(\phi_x^+, 0)^2), & \text{se } \phi \leq 0. \end{cases} \quad (95)$$

Conforme descrito a seguir, a equação de reinicialização utiliza apenas os valores de  $\phi_x^2$ , de forma que não é necessário determinar os valores de  $\phi_x$ , e a aproximação descrita pela Eq. 95 pode ser utilizada de forma direta.

Com as aproximações para  $\phi_x^2$ ,  $\phi_y^2$  e  $\phi_z^2$  já definidas, o resto da discretização é bastante direto. De forma análoga ao desenvolvimento apresentado para a equação de transporte do Level Set, uma aplicação do método de Euler explícito à equação de reinicialização (Eq. 41) é realizada da forma:

$$\phi_{ijk}^{n+1} = \phi_{ijk}^n + \Delta\tau S(\phi_{ijk}^n) \left( 1 - \sqrt{\frac{\partial\phi^{n^2}}{\partial x} + \frac{\partial\phi^{n^2}}{\partial y} + \frac{\partial\phi^{n^2}}{\partial z}} \right), \quad (96)$$

em que a função sinal é calculada de acordo com a definição da Eq. 46, com as mesmas

aproximações já definidas para as derivadas de  $\phi$ , na forma:

$$S(\phi) = \frac{\phi}{\sqrt{\phi^2 + (\phi_x^2 + \phi_y^2 + \phi_z^2) \max(\Delta x, \Delta y, \Delta z)^2}}. \quad (97)$$

### 2.7.7 Condições de contorno para o problema da cavidade

Para aplicar as condições de contorno, são utilizados *ghosts*, que são pontos "imaginários" da malha, situados fora do domínio. Estes pontos seguem a mesma organização dos demais pontos da malha, e têm seus valores calculados de acordo com as condições de contorno que se deseja aplicar. Para os casos em que nós da malha coincidem com as paredes da cavidade, a utilização de *ghosts* não é necessária. Nesta seção serão exemplificadas as aplicações de condições de contorno para  $u$ ,  $u^*$ , e  $\chi$ , com a aplicação das condições de contorno para as demais componentes da velocidade sendo realizada de forma análoga. Nota-se que as condições de contorno são aplicadas a todas as variáveis do escoamento, e não apenas às variáveis calculadas por sistemas lineares. Isto é necessário, por exemplo, para o cálculo correto do termo  $\mathbf{u} \cdot \nabla \mathbf{u}$ , realizado ao final de cada iteração (ver seção 2.10). Estas condições de contorno são as condições de aderência e impenetrabilidade para as velocidades (Dirichlet), e de Neumann homogêneas para  $P$  e  $\chi$ . Para os casos em que nenhum dos nós de uma malha de velocidade coincide com a parede, é utilizada interpolação linear entre os dois pontos adjacentes da malha (um deles um *ghost*) para calcular o valor da velocidade na parede.

Para o caso das componentes de velocidade normais à uma superfície, os nós da malha desta componente coincidem com a parede, de modo que, utilizando a Eq. 20, temos que o valor do *ghost* será igual ao valor da velocidade na parede, determinado pela condição de contorno, e não dependerá da resolução de sistemas lineares. Já para o caso das componentes tangenciais, é necessário utilizar uma interpolação linear para encontrar o valor da componente na parede, bem como utilizar a Eq. 21 para relacionar os valores de  $\mathbf{u}^*$  e  $\mathbf{u}$ . Como exemplo, para a parede  $y = 0$ , e utilizando a Eq. 18, temos que

$$\frac{u_{i0k} + u_{i1k}}{2} = 0 \Rightarrow u_{i0k}^* = 2 \left( 0 + \Delta t \frac{\chi_{i+10k}^{n+1} - \chi_{i0k}^{n+1}}{\Delta x} \right) - u_{i1k}^*. \quad (98)$$

Nota-se que, na Eq. 98, o valor de  $u_{i0k}^*$  depende de  $u_{i1k}^*$ . Entretanto, para o sistema linear na forma de estêncil, o parâmetro  $b_{i1k}$  deve multiplicar apenas  $u_{i0k}^*$ , ao passo que  $u_{i1k}^*$  deve ser multiplicado apenas pelo parâmetro  $d_{i1k}$ . Deste modo, é necessário alterar os valores dos parâmetros do sistema linear,

$$d_{i1k} \Leftarrow d_{i1k} - b_{i1k}, \quad (99)$$

ao passo que o valor de  $u_{i0k}^*$ , a ser multiplicado por  $b_{i1k}$ , se torna

$$u_{i0k}^* = 2 \left( 0 + \Delta t \frac{\chi_{i+10k}^{n+1} - \chi_{i0k}^{n+1}}{\Delta x} \right). \quad (100)$$

Os valores dos *ghosts* de  $u^*$  para as demais paredes, bem como os valores demonstrados para a parede  $y = 0$ , estão dispostos na Tab. 1.

Parede	Valor dos <i>ghosts</i>
$x = 0$	$u_{0,j,k}^* = 0$
$x = 1$	$u_{NX+1,j,k}^* = 0$
$y = 0$	$u_{i,0,k}^* = 2\{0 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,1,k}^n - \chi_{i,1,k}^n) - (\chi_{i+1,1,k}^{n-1} - \chi_{i,1,k}^{n-1})]\}$
$y = \Gamma$	$u_{i,NY+1,k}^* = 2\{1 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,NY,k}^n - \chi_{i,NY,k}^n) - (\chi_{i+1,NY,k}^{n-1} - \chi_{i,NY,k}^{n-1})]\}$
$z = 0$	$u_{i,j,0}^* = 2\{0 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,j,1}^n - \chi_{i,j,1}^n) - (\chi_{i+1,j,1}^{n-1} - \chi_{i,j,1}^{n-1})]\}$
$z = \Lambda$	$u_{i,j,NZ+1}^* = 2\{0 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,j,NZ}^n - \chi_{i,j,NZ}^n) - (\chi_{i+1,j,NZ}^{n-1} - \chi_{i,j,NZ}^{n-1})]\}$

Tabela 1 – Valores dos *ghosts* de  $u^*$  para o problema da cavidade

A Tab. 2 apresenta os valores dos *ghosts* de  $u$  e  $\chi$ , respectivamente. Os valores dos *ghosts* de  $u$  são determinados de forma trivial com a utilização de interpolação linear, ao passo que os *ghosts* de  $\chi$  são facilmente calculados com a utilização da Eq. 18. Da mesma forma como ocorre com  $u^*$ , também é necessário alterar os valores dos parâmetros do sistema linear de  $\chi$ .

Parede	Valor dos <i>ghosts</i> de $u$	Valor dos <i>ghosts</i> de $\chi$
$x = 0$	$u_{0,j,k} = 0$	$\chi_{0,j,k} = \chi_{1,j,k}$
$x = 1$	$u_{NX+1,j,k} = 0$	$\chi_{NX+1,j,k} = \chi_{NX,j,k}$
$y = 0$	$u_{i,0,k} = 0 - u_{i,1,k}$	$\chi_{i,0,k} = \chi_{i,1,k}$
$y = \Gamma$	$u_{i,NY+1,k} = 2 - u_{i,NY,k}$	$\chi_{i,NY+1,k} = \chi_{i,NY,k}$
$z = 0$	$u_{i,j,0} = 0 - u_{i,j,1}$	$\chi_{i,j,0} = \chi_{i,j,1}$
$z = \Lambda$	$u_{i,j,NZ+1} = 0 - u_{i,j,NZ}$	$\chi_{i,j,NZ+1} = \chi_{i,j,NZ}$

Tabela 2 – Valores dos *ghosts* de  $u$  e  $\chi$  para o problema da cavidade

As alterações realizadas nos parâmetros dos sistemas lineares na forma de estêncil, de acordo com os valores aplicados aos *ghosts*, estão dispostas na Tab. 3, para todas as variáveis calculadas pela resolução de sistemas lineares.

### 2.7.8 Condições de contorno para cisalhamento simples

Para simulações de escoamentos de cisalhamento simples, é necessário alterar as condições de contorno já apresentadas. Para o caso das paredes normais ao eixo  $y$ , a aplicação das condições de contorno é feita de forma idêntica, apenas com a alteração

Parede	$u^*$	$v^*$	$w^*$	$\chi$
$x = 0$	-	$d \leftarrow d - a$	$d \leftarrow d - a$	$d \leftarrow d + a$ $a \leftarrow 0$
$x = 1$	-	$d \leftarrow d - e$	$d \leftarrow d - e$	$d \leftarrow d + e$ $e \leftarrow 0$
$y = 0$	$d \leftarrow d - b$	-	$d \leftarrow d - b$	$d \leftarrow d + b$ $b \leftarrow 0$
$y = \Gamma$	$d \leftarrow d - f$	-	$d \leftarrow d - f$	$d \leftarrow d + f$ $f \leftarrow 0$
$z = 0$	$d \leftarrow d - c$	$d \leftarrow d - c$	-	$d \leftarrow d + c$ $c \leftarrow 0$
$z = \Lambda$	$d \leftarrow d - g$	$d \leftarrow d - g$	-	$d \leftarrow d + g$ $g \leftarrow 0$

Tabela 3 – Alterações dos parâmetros dos sistemas lineares no contorno, para o caso do problema da cavidade

das velocidades de movimentação das paredes. Para as outras fronteiras, entretanto, a aplicação das condições de contorno é feita de forma completamente diferente, tendo em vista que as condições de contorno são periódicas. Isto significa que, para uma dada fronteira, todas as propriedades do escoamento devem ser idênticas às propriedades na fronteira oposta, incluindo suas respectivas derivadas. Para o caso da discretização aqui apresentada, isto é feito atribuindo valores aos *ghosts* iguais aos valores dos nós internos adjacentes à fronteira oposta. Por exemplo:

$$\begin{cases} u_{0jk} = u_{NXjk} \\ u_{NX+1jk} = u_{1jk} . \end{cases} \quad (101)$$

É importante notar que, nas paredes em que as condições de contorno são periódicas, não é necessário alterar os valores dos parâmetros dos sistemas lineares ( $a, b, c, d, e, f$  e  $g$ ), nem atribuir valores de *ghosts* para a resolução correta destes sistemas lineares. Isto ocorre porque o próprio método de solução dos sistemas lineares é alterado para aplicar as condições de contorno periódicas, tendo em vista que, caso fossem utilizados *ghosts*, estes deveriam variar continuamente durante o processo de solução dos sistemas lineares.

As condições de contorno aplicadas para o caso de cisalhamento simples estão dispostas nas Tabs. 4, 5 e 6.

## 2.8 O MÉTODO DOS GRADIENTES CONJUGADOS

Para resolver os sistemas lineares, o código desenvolvido utiliza o método dos Gradientes Conjugados com um preconditionador do tipo SSOR. Em versões futuras deste código, é possível que sejam empregados métodos diferentes para resolução de sistemas

Parede	Valor dos <i>ghosts</i>
$x = 0$	-
$x = L_x$	-
$y = 0$	$u_{i,0,k}^* = 2\{-L_y/2 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,1,k}^n - \chi_{i,1,k}^n) - (\chi_{i+1,1,k}^{n-1} - \chi_{i,1,k}^{n-1})]\}$
$y = L_y$	$u_{i,NY+1,k}^* = 2\{L_y/2 + \frac{\Delta t}{\Delta x}[2(\chi_{i+1,NY,k}^n - \chi_{i,NY,k}^n) - (\chi_{i+1,NY,k}^{n-1} - \chi_{i,NY,k}^{n-1})]\}$
$z = 0$	-
$z = L_z$	-

Tabela 4 – Valores dos *ghosts* de  $u^*$  para o caso de cisalhamento simples

Parede	Valor dos <i>ghosts</i> de $u$	Valor dos <i>ghosts</i> de $\chi$
$x = 0$	$u_{0,j,k} = u_{NX,j,k}$	$\chi_{0,j,k} = \chi_{NX,j,k}$
$x = L_x$	$u_{NX+1,j,k} = u_{1,j,k}$	$\chi_{NX+1,j,k} = \chi_{1,j,k}$
$y = 0$	$u_{i,0,k} = u_{i,NY,k}$	$\chi_{i,0,k} = \chi_{i,NY,k}$
$y = L_y$	$u_{i,NY+1,k} = u_{i,1,k}$	$\chi_{i,NY+1,k} = \chi_{i,1,k}$
$z = 0$	$u_{i,j,0} = u_{i,j,NZ}$	$\chi_{i,j,0} = \chi_{i,j,NZ}$
$z = L_z$	$u_{i,j,NZ+1} = u_{i,j,1}$	$\chi_{i,j,NZ+1} = \chi_{i,j,1}$

Tabela 5 – Valores dos *ghosts* de  $u$  e  $\chi$  para o caso de cisalhamento simples

Parede	$u^*$	$v^*$	$w^*$	$\chi$
$x = 0$	-	-	-	-
$x = L_x$	-	-	-	-
$y = 0$	$d \leftarrow d - b$	-	$d \leftarrow d - b$	$d \leftarrow d + b$ $b \leftarrow 0$
$y = L_y$	$d \leftarrow d - f$	-	$d \leftarrow d - f$	$d \leftarrow d + f$ $f \leftarrow 0$
$z = 0$	-	-	-	-
$z = L_z$	-	-	-	-

Tabela 6 – Alterações dos parâmetros dos sistemas lineares no contorno, para o caso de cisalhamento simples



lineares, como por exemplo métodos *multigrid*, ou a utilização de *Fast Fourier Solvers* para a equação da pressão, de modo a aumentar a velocidade de solução das simulações.

Para um sistema linear genérico na forma  $A \cdot \mathbf{x} = \mathbf{b}$ , o algoritmo de Gradientes Conjugados com preconditionamento utilizado, conforme apresentado por Shewchuk et al. (1994), é:

Etapa	Operação
1.	$\mathbf{r}^0 = \mathbf{b} - A \cdot \mathbf{x}$
2.	$\mathbf{z}^0 = M^{-1} \cdot \mathbf{r}^0$
3.	$\mathbf{d}^0 = \mathbf{z}^0$
4.	$\alpha = \frac{\mathbf{r}^i \cdot \mathbf{z}^i}{\mathbf{d}^i \cdot A \cdot \mathbf{d}^i}$
5.	$\mathbf{x}^{i+1} = \mathbf{x}^i + \alpha \mathbf{d}^i$
6.	$\mathbf{r}^{i+1} = \mathbf{r}^i - \alpha A \cdot \mathbf{d}^i$
7.	$\mathbf{z}^{i+1} = M^{-1} \cdot \mathbf{r}^{i+1}$
8.	$\beta = \frac{\mathbf{r}^{i+1} \cdot \mathbf{z}^{i+1}}{\mathbf{r}^i \cdot \mathbf{z}^i}$
9.	$\mathbf{d}^{i+1} = \mathbf{z}^{i+1} + \beta \mathbf{d}^i$

Tabela 7 – Algoritmo de Gradientes Conjugados com preconditionamento (SHEWCHUK et al., 1994)

Inicia-se o método em  $i = 0$ , com um chute inicial  $x^0$ . Após a primeira iteração, repetem-se os passos 4-9, até que a solução convirja para um resíduo,  $\mathbf{r}$ , menor que a tolerância desejada.

Para um preconditionamento por SSOR, a matriz preconditionadora  $M$  é definida, de acordo com Saad (2003), como

$$M_{SSOR} = \frac{1}{\omega(2 - \omega)}(D + \omega L)D^{-1}(D + \omega L^T), \quad (102)$$

em que  $D$  é a diagonal da matriz  $A$ ,  $L$  é a matriz triangular inferior de  $A$ , e  $\omega$  é um parâmetro de relaxação. Como a matriz  $A$  deve ser simétrica para o método SSOR, temos que

$$A = L + D + L^T. \quad (103)$$

Por fim, o valor ótimo para o parâmetro de relaxação,  $\omega_{opt}$ , é definido como

$$\omega_{opt} = \frac{2}{1 + \sin(\pi h)}, \quad (104)$$

conforme demonstrado por Yang e Gobbert (2009), em que

$$h = \min(\Delta x, \Delta y, \Delta z). \quad (105)$$

Como os sistemas lineares resolvidos neste trabalho são definidos na forma de estêncil, e não na forma  $A \cdot \mathbf{x} = \mathbf{b}$ , é necessário traçar paralelos entre as definições, de modo a implementar o método dos Gradientes Conjugados. Para isso, é necessário definir as operações de multiplicação entre matriz e vetor, e de produto escalar entre dois vetores. Para um produto entre matriz e vetor na forma  $\mathbf{p} = A \cdot \mathbf{x}$ , este pode ser escrito na forma de estêncil como

$$p_{ijk} = a_{ijk}x_{i-1jk} + b_{ijk}x_{ij-1k} + c_{ijk}x_{ijk-1} + d_{ijk}x_{ijk} + e_{ijk}x_{i+1jk} + f_{ijk}x_{ij+1k} + g_{ijk}x_{ijk+1}. \quad (106)$$

Para o caso do problema da cavidade, esta definição é utilizada sem alterações em todos os pontos da malha, já que são utilizados *ghosts* e alterações nos parâmetros de  $A$  para aplicar as condições de contorno. Para o caso de cisalhamento simples, entretanto, as condições de contorno periódicas não podem ser aplicadas desta forma. Deste modo, a definição do produto  $A \cdot \mathbf{x}$  é alterada para os nós adjacentes às fronteiras, aplicando as condições de contorno de que  $x_{0jk} = x_{NXjk}$ ,  $x_{NX+1jk} = x_{1jk}$ ,  $x_{ij0} = x_{ijNZ}$  e  $x_{ijNZ+1} = x_{ij1}$ .

Já o produto escalar entre dois vetores quaisquer, na forma de estêncil, pode ser definido como

$$\mathbf{a} \cdot \mathbf{b} = \sum_{k=1}^{NZ} \sum_{j=1}^{NY} \sum_{i=1}^{NX} a_{ijk} b_{ijk}. \quad (107)$$

Para utilizar o condicionamento, também é necessário determinar uma forma de calcular  $\mathbf{z}$ . Este cálculo é feito de forma direta, ou seja, não é necessário montar a matriz condicionadora  $M$ . A dedução do cálculo de  $\mathbf{z}$  é feita da seguinte maneira:

$$\mathbf{z} = M^{-1} \cdot \mathbf{r} \Rightarrow M \cdot \mathbf{z} = \mathbf{r}. \quad (108)$$

Para um condicionamento com SSOR:

$$\frac{1}{\omega(2-\omega)}(D + \omega L)D^{-1} \cdot \mathbf{z}' = \mathbf{r}, \quad (109)$$

$$\mathbf{z}' = (D + \omega L^T) \cdot \mathbf{z}, \quad (110)$$

$$\mathbf{z}' + \omega L D^{-1} \cdot \mathbf{z}' = \omega(2-\omega)\mathbf{r}. \quad (111)$$

Passando o segundo termo do lado esquerdo da Eq. 111 para o lado direito, e reescrevendo na forma de estêncil,

$$z'_{ijk} = \omega(2-\omega)r_{ijk} - \frac{\omega}{c_{ijk}}(a_{ijk}z'_{i-1jk} + b_{ijk}z'_{ij-1k} + c_{ijk}z'_{ijk-1}). \quad (112)$$

Nota-se que a Eq. 112 deve ser resolvida para todos os pontos da malha, com  $i$ ,  $j$  e  $k$  crescentes. Deste modo, os valores de  $\mathbf{z}'$  presentes no lado direito da equação já terão sido calculados previamente.

Para determinar o valor de  $z$ , utiliza-se a Eq. 110:

$$D \cdot \mathbf{z} + \omega L^T \cdot \mathbf{z} = \mathbf{z}' \Rightarrow \mathbf{z} = D^{-1}(\mathbf{z}' - \omega L^T \cdot \mathbf{z}). \quad (113)$$

Reescrevendo na forma de estêncil, temos que

$$z_{ijk} = \frac{z'_{ijk} - \omega(e_{ijk}z_{i+1jk} + f_{ijk}z_{ij+1k} + g_{ijk}z_{ijk+1})}{c_{ijk}}. \quad (114)$$

Nota-se que a Eq. 114 deve ser resolvida para todos os pontos da malha, com  $i$ ,  $j$  e  $k$  decrescentes, de modo que os valores de  $z$  do lado direito da equação já tenham sido calculados previamente.

## 2.9 PÓS-PROCESSAMENTO DOS RESULTADOS

Para o caso de simulações de escoamentos bifásicos, é necessário determinar uma série de medidas para a gota, de forma a permitir uma correta validação do código computacional desenvolvido. Essas medidas são de volume, para garantir a conservação de massa da gota (densidade constante), inclinação e deformação da gota, e o cálculo do tensor *Stresslet*, utilizado para determinar as tensões induzidas no escoamento pela presença da gota, de modo a permitir uma descrição do comportamento de uma emulsão, sem a necessidade de se simular individualmente o comportamento de cada gota. O cálculo de volume da gota é feito com a utilização da Eq. 39 e da função Heaviside:

$$V_g = \int_V H(-\phi) dV, \quad (115)$$

em que  $V_g$  é o volume da gota,  $V$  representa todo o domínio computacional, e  $dV$  pode ser descrito como  $\Delta x \Delta y \Delta z$ .

Para medir a inclinação e deformação da gota, é utilizado o tensor de inércia  $I$ . Com a utilização de notação de Einstein, este tensor pode ser descrito como

$$I_{ij} = \int_V (x_k^2 \delta_{ij} - x_i x_j) H(-\phi) dV, \quad (116)$$

em que  $\delta_{ij}$  é o delta de Kronecker. Uma propriedade importante do tensor de inércia é que seus três autovalores correspondem aos três momentos principais de inércia, e os autovetores correspondentes a estes autovalores representam os três eixos principais de rotação, associados a cada um dos momentos principais de inércia. Deste modo, a direção de inclinação da gota é representada pelo autovetor associado ao menor dos autovalores, já que este autovalor será correspondente ao eixo longo da gota. Os cálculos dos autovalores e autovetores são feitos com a utilização do algoritmo escrito por Kopp (2008).

Para se calcular a deformação da gota, no caso de pequenas deformações, é utilizada a hipótese de que o formato da gota é uma elipsoide. Deste modo, como a gota

possui densidade constante, os três momentos principais de inércia de um elipsoide são função unicamente da massa (volume) da gota, e dos comprimentos associados aos três eixos principais de rotação. Como os três momentos principais de inércia e o volume já são conhecidos, é possível calcular os três comprimentos.

Com as dimensões da gota conhecidas, é possível calcular a deformação de Taylor. Esta medida de deformação, introduzida por Taylor (1934), é definida como

$$D_T = \frac{L - B}{L + B}, \quad (117)$$

em que  $L$  é o comprimento da gota e  $B$  a altura, ambos medidos no plano de cisalhamento do escoamento. Nota-se que a deformação de Taylor é igual a zero para o caso em que  $L = B$  (gota esférica), e converge para 1 quando  $L \gg B$ .

## 2.10 DESCRIÇÃO GERAL DO PROGRAMA COMPLETO

Com todas as componentes do código devidamente apresentadas, incluindo a apresentação das equações governantes, bem como suas discretizações tanto no tempo quanto no espaço, e a aplicação de condições de contorno, resta apenas organizar todas estas etapas na forma de um programa computacional. Primeiramente, o programa possui uma etapa de inicialização, com leitura do arquivo de entrada, que especifica as características de escoamento e de malha desejados, além de alocar e inicializar todas as variáveis globais, incluindo a função Level Set, e os parâmetros fixos dos sistemas lineares  $(a, b, c, d, e, f, g)$ , já com as devidas condições de contorno aplicadas. Caso o escoamento a ser simulado seja bifásico, também é executada a sub-rotina do método de Level Set local, para determinar quais pontos da malha estão situados nas regiões em que serão realizadas as operações com o Level Set, e planificar a função Level Set nos pontos situados fora destas regiões.

Após este processo de inicialização, o programa entra em sua parte iterativa, que consiste em:

1. Atualizar os valores dos *ghosts* de  $u^*, v^*, w^*$  e  $\chi$ , de acordo com as condições de contorno utilizadas.
2. Atualizar os valores dos parâmetros variáveis ( $h$ ) dos sistemas lineares associados às velocidades-tentativa  $(u^*, v^*, w^*)$ . Após esta etapa, estes sistemas lineares estão totalmente definidos.
3. Utilização do método de Gradientes Conjugados para resolver os sistemas lineares já definidos, de modo a determinar os novos valores das variáveis  $u^*, v^*$  e  $w^*$ .
4. Atualizar os valores do parâmetro variável ( $h$ ) do sistema linear associado à variável auxiliar de pressão  $\chi$ . Após esta etapa, este sistema linear está totalmente definido.

5. Utilização do método de Gradientes Conjugados para resolver o sistema linear definido anteriormente, de modo a determinar os novos valores da variável auxiliar de pressão  $\chi$ .
6. Calcular os valores das velocidades reais  $u, v$  e  $w$ , bem como atualizar os valores de seus respectivos *ghosts*, de acordo com as condições de contorno utilizadas.
7. Calcular os valores dos termos convectivos das equações de Navier-Stokes, isto é, os termos relativos à  $\mathbf{u} \cdot \nabla \mathbf{u}$ .
8. Calcular os valores da pressão real  $P$ .
9. (Escoamento bifásico) Realizar o transporte da função Level Set.
10. (Escoamento bifásico, iterações seletas) Realizar a operação de reinicialização do Level Set, caso seja uma das iterações em que esta operação deva ocorrer.
11. (Escoamento bifásico) Executar a sub-rotina do método de Level Set local, para determinar quais pontos da malha estão situados nas regiões em que serão realizadas as operações com o Level Set, e planificar a função Level Set nos pontos situados fora destas regiões.

Ocasionalmente, ao final das iterações, são executadas rotinas de pós-processamento dos dados, de modo a calcular a inclinação e deformação da gota, e de exportação de resultados.

## 3 RESULTADOS

### 3.1 ESCOAMENTOS MONOFÁSICOS NA CAVIDADE

#### 3.1.1 Verificação de ordem do método

A primeira etapa para validação do código numérico desenvolvido é a verificação de ordem, tanto para a discretização espacial quanto para a discretização temporal. Isto também permitirá uma escolha mais consciente para as malhas a serem utilizadas nas demais simulações. Como apresentado na seção acerca da metodologia, é esperado que o código desenvolvido seja de segunda ordem, tanto para a discretização espacial quanto para a discretização temporal.

Para realizar a análise de ordem da discretização espacial, foram realizadas simulações com diferentes refinamentos de malha, para o problema da cavidade com um número de Reynolds de valor intermediário,  $Re = 300$ , e uma cavidade cúbica. As malhas utilizadas foram de  $20 \times 20 \times 20$ ,  $30 \times 30 \times 30$ ,  $50 \times 50 \times 50$ ,  $80 \times 80 \times 80$ ,  $110 \times 110 \times 110$ , e  $170 \times 170 \times 170$ , com passo de tempo sempre constante, de  $\Delta t = 2 \times 10^{-3}$ . Para o caso da malha mais refinada, temos que o número de CFL para a condição crítica, nas proximidades da parede superior, é:

$$CFL = \frac{U\Delta t}{\Delta x} = \frac{2 \times 10^{-3}}{1/170} = 0,34. \quad (118)$$

As simulações foram realizadas apenas até o instante  $t = 1$ , de modo a economizar recursos computacionais. Nota-se que, neste instante, o escoamento ainda não se encontra em regime permanente.

A Fig. 5 apresenta um gráfico em escala logarítmica da norma do erro associado à cada uma das simulações, em função das distâncias entre os nós das respectivas malhas ( $\Delta x = \Delta y = \Delta z$ ). Para calcular a norma do erro, foram utilizados os valores da velocidade  $u$  sobre o plano definido por  $x = 0,5$ , com os resultados obtidos para a malha de  $170 \times 170 \times 170$  sendo utilizados como valor de referência, já que uma solução exata não está disponível. Neste gráfico, nota-se que a evolução da norma do erro é bastante próxima de uma proporcionalidade com  $\Delta x^2$ , o que está de acordo com a ordem 2 esperada para o método desenvolvido.

Já para a análise de ordem da discretização temporal, também foram realizadas simulações em uma cavidade cúbica com  $Re = 300$ , porém desta vez com variações do passo de tempo. Entretanto, de modo a evitar que erros provenientes da discretização espacial se sobreponham aos erros provenientes da discretização temporal, foram utilizadas

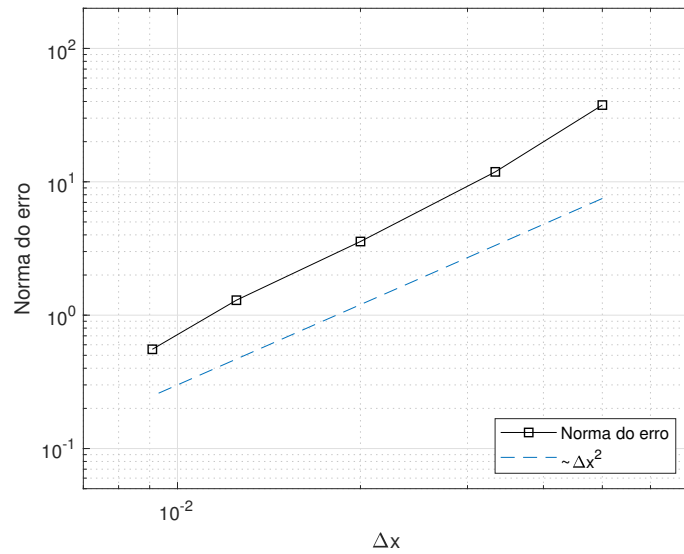


Figura 5 – Análise de ordem da discretização espacial

malhas progressivamente mais refinadas com a redução no passo de tempo. Isto foi feito com base no número de CFL, mantido constante como  $1/6$ . Como a velocidade  $U = 1$ , isto significa que os passos de tempo foram determinados como  $\Delta t = \Delta x/6$ .

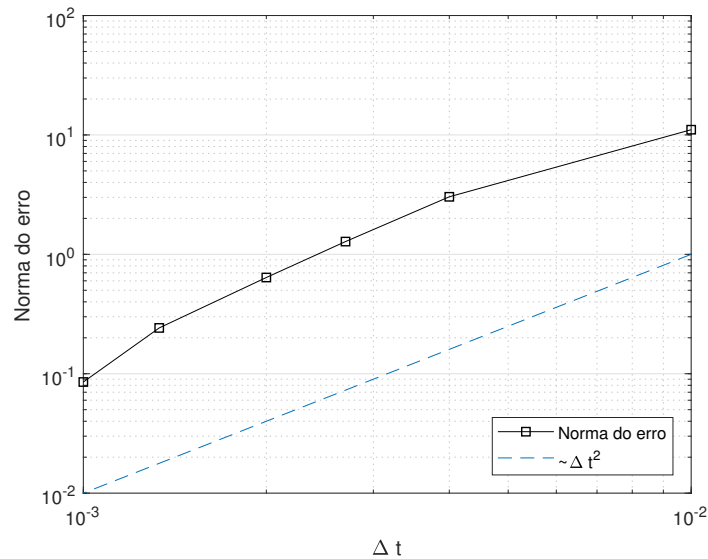


Figura 6 – Análise de ordem da discretização temporal

A Fig. 6 apresenta um gráfico em escala logarítmica da norma do erro em função do passo de tempo utilizado. A norma do erro foi calculada com base nos valores de  $u$  no plano definido por  $x = 0,5$ . Nota-se que esta análise indica que o código computacional criado é de segunda ordem no tempo, de acordo com o esperado. Deste modo, fica comprovada a ordem 2 das aproximações para as velocidades utilizadas neste método, tanto para a discretização espacial quanto para a discretização temporal.

### 3.1.2 Comparação com resultados disponíveis na literatura

Para verificar a acurácia do código numérico desenvolvido, é de suma importância que seja feita uma comparação entre os resultados obtidos e resultados disponíveis na literatura. De fato, validação da parte monofásica do código numérico é o único motivo para o qual foram realizadas simulações do problema da cavidade. Para tal, os resultados obtidos serão comparados com os resultados apresentados por Altensoeder e Kuhlmann (2005), para números de Reynolds iguais a 100 e 1000, e diferentes dimensões da cavidade.

O primeiro caso trata de uma cavidade cúbica, isto é,  $L_y = L_z = 1$ , com  $Re = 100$ . Foi utilizada uma malha de  $110 \times 110 \times 110$ , com  $\Delta t = 2 \times 10^{-3}$ , e os resultados apresentados correspondem ao instante  $t = 30$ . Deste modo, o número de  $CFL$  para o caso extremo, na parede superior, é:

$$CFL = \frac{U \Delta t}{\Delta x} = \frac{2 \times 10^{-3}}{1/110} = 0,22. \quad (119)$$

Para comparar os resultados obtidos com aqueles apresentados por Altensoeder e Kuhlmann (2005), são traçados perfis das velocidades  $u$  e  $v$  no centro da cavidade, dispostos nas Figs. 7 e 8, respectivamente.

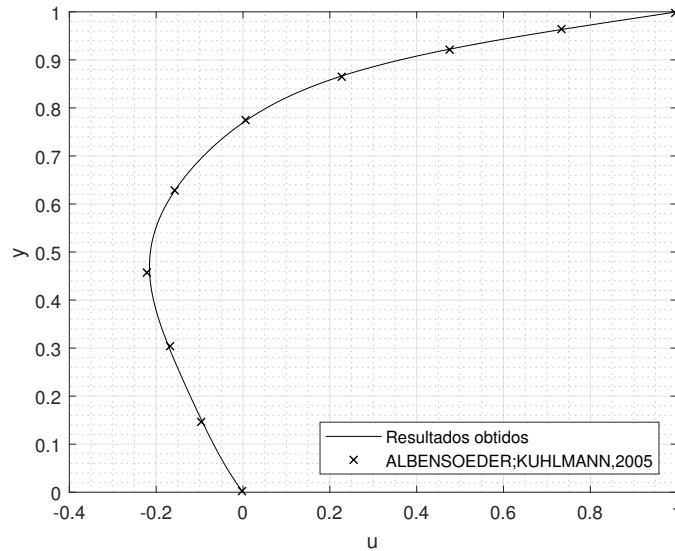


Figura 7 – Perfil da velocidade  $u$  para  $x = 0,5$  e  $z = 0,5$ .  $Re = 100$ ,  $L_y = 1$  e  $L_z = 1$

Nota-se que os resultados obtidos são idênticos aos apresentados por Altensoeder e Kuhlmann (2005).

Para o segundo caso, os parâmetros utilizados são idênticos aos do primeiro caso, com exceção do número de Reynolds, aumentado para 1000, e o tempo final, de  $t = 60$ . Novamente, são traçados perfis das velocidades  $u$  e  $v$  no centro da cavidade, dispostos nas Figs. 9 e 10, respectivamente. Também é traçado um perfil da componente  $w$  da velocidade, na Fig. 15.



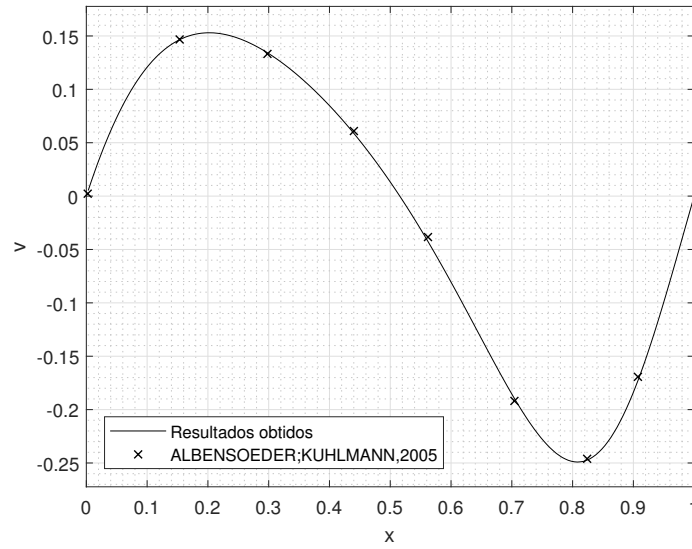


Figura 8 – Perfil da velocidade  $v$  para  $y = 0,5$  e  $z = 0,5$ .  $Re = 100$ ,  $L_y = 1$  e  $L_z = 1$

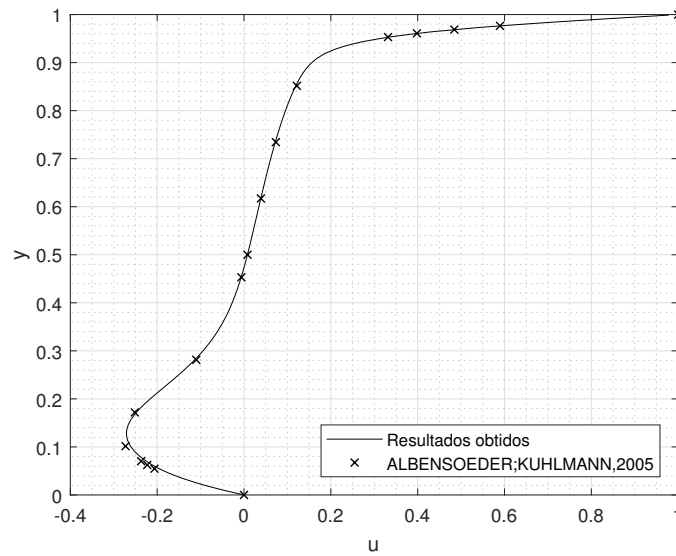


Figura 9 – Perfil da velocidade  $u$  para  $x = 0,5$  e  $z = 0,5$ .  $Re = 1000$ ,  $L_y = 1$  e  $L_z = 1$

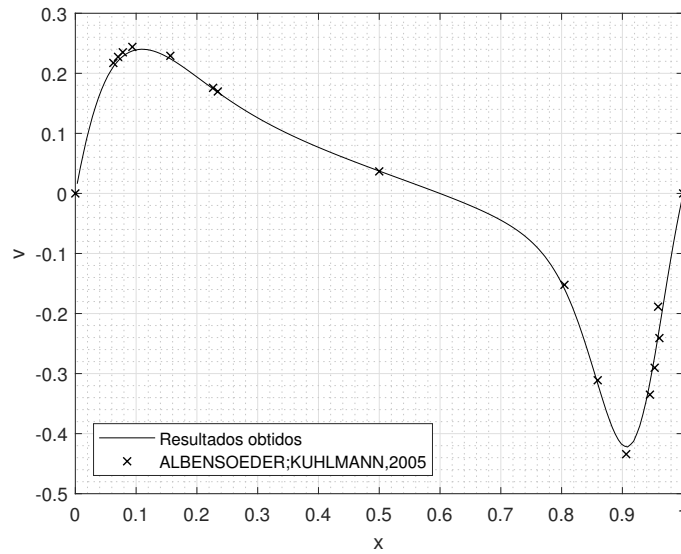


Figura 10 – Perfil da velocidade  $v$  para  $y = 0,5$  e  $z = 0,5$ .  $Re = 1000$ ,  $L_y = 1$  e  $L_z = 1$

Mesmo com o aumento do número de Reynolds, os resultados obtidos ainda se mostram quase idênticos aos apresentados Albensoeder e Kuhlmann (2005).

A seguir, são apresentados resultados para uma cavidade mais profunda, com  $L_z = 2$ . Os valores de  $Re$  e  $L_y$  são mantidos em 1000 e 1, respectivamente. Foi utilizada uma malha de  $100 \times 100 \times 150$ , com intuito de manter algum refinamento da malha na direção  $z$ , sem tornar o custo computacional inviável. Foram utilizados os mesmos valores de  $\Delta t = 2 \times 10^{-3}$ , com a simulação indo até o instante  $t = 60$ . Estes resultados estão dispostos nas Figs. 11 e 12, com perfis das componentes  $u$  e  $v$  da velocidade, respectivamente. Um perfil da componente  $w$  da velocidade também é traçado na Fig. 15.

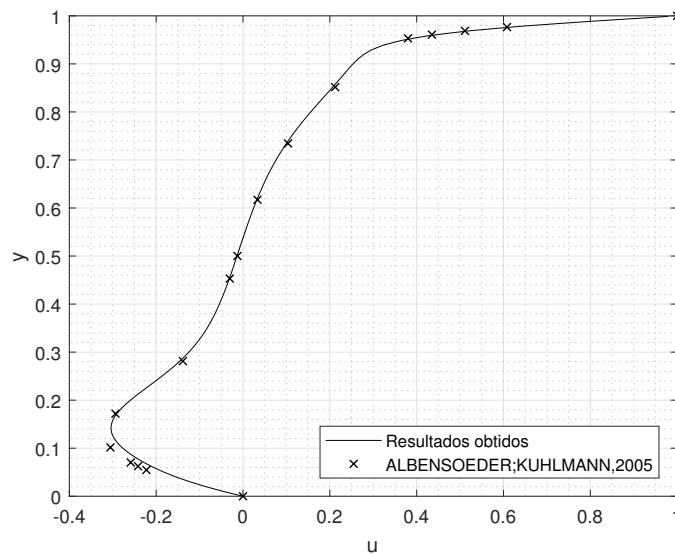


Figura 11 – Perfil da velocidade  $u$  para  $x = 0,5$  e  $z = 1$ .  $Re = 1000$ ,  $L_y = 1$  e  $L_z = 2$

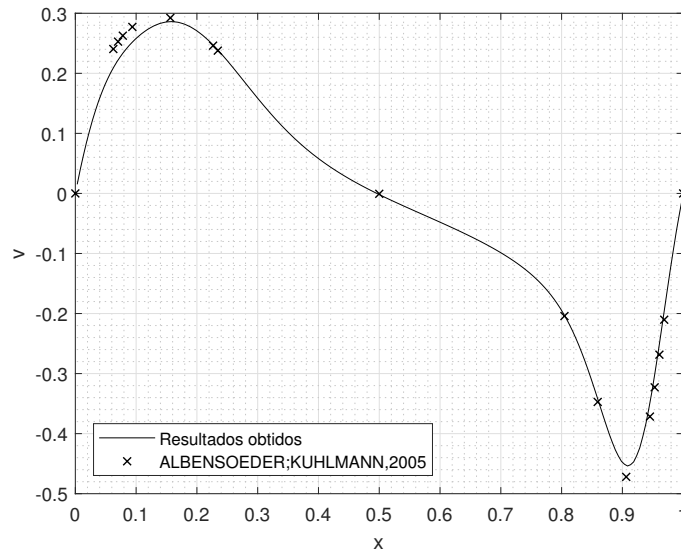


Figura 12 – Perfil da velocidade  $v$  para  $y = 0,5$  e  $z = 1$ .  $Re = 1000$ ,  $L_y = 1$  e  $L_z = 2$

Nota-se que, apesar de os resultados ainda se mostrarem bastante próximos dos resultados de referência apresentados por Albensoeder e Kuhlmann (2005), aparecem algumas pequenas discrepâncias nos extremos do perfil da velocidade  $v$ .

O último caso estudado para validação do código monofásico, é o caso de uma cavidade mais alta, isto é, com  $L_y = 1$ . Os valores de  $L_z$  e  $Re$  utilizados são de 1 e 1000, respectivamente, com uma malha de  $100 \times 150 \times 100$ , de modo a manter algum refinamento na direção  $y$  sem tornar o custo computacional excessivamente elevado. Foi utilizado  $\Delta t = 2 \times 10^{-3}$ , indo até o instante  $t = 60$ .

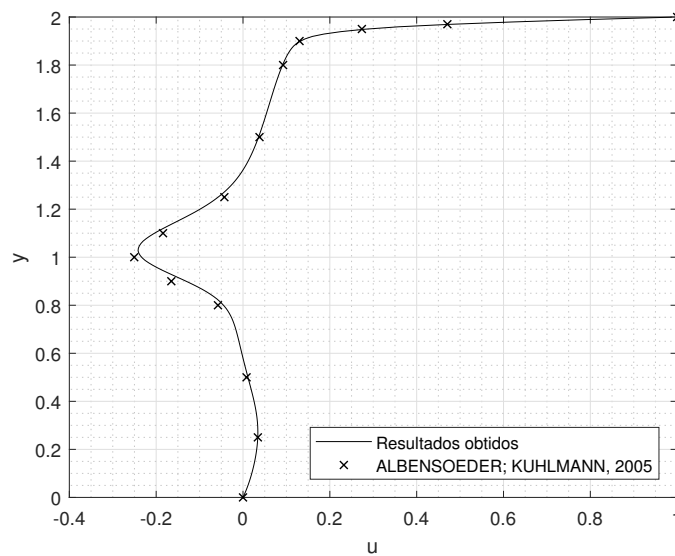


Figura 13 – Perfil da velocidade  $u$  para  $x = 0,5$  e  $z = 0,5$ .  $Re = 1000$ ,  $L_y = 2$  e  $L_z = 1$

As Figs. 13 e 14 apresentam perfis das componentes de velocidade  $u$  e  $v$ , respec-

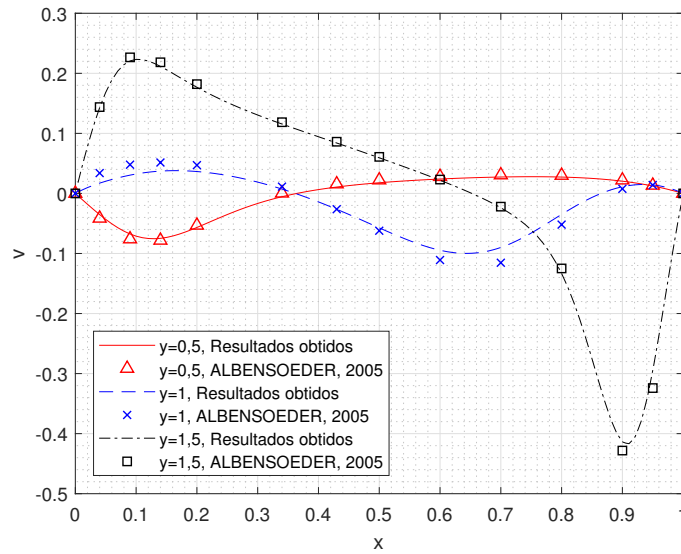


Figura 14 – Perfis da velocidade  $v$  para  $y = 0,5$ ,  $y = 1$  e  $y = 1,5$ , com  $z = 0,5$ .  $Re = 1000$ ,  $L_y = 2$  e  $L_z = 1$

tivamente, para esta cavidade mais alta. Nota-se uma concordância muito boa entre os resultados obtidos e os valores utilizados como referência, apresentados por Albensoeder e Kuhlmann (2005), com as discrepâncias observadas sendo pequenas e pontuais.

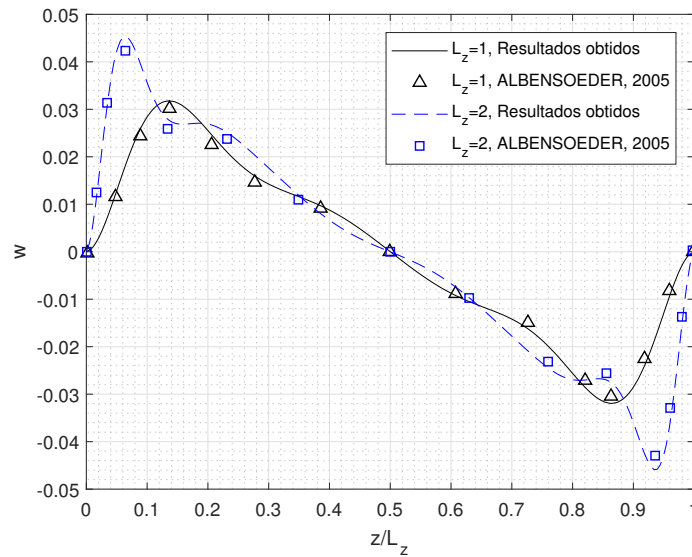


Figura 15 – Perfil da velocidade  $w$  para  $x = 0,5$  e  $y = 0,5$ .  $Re = 1000$ ,  $L_y = 1$

De modo geral, os resultados obtidos estão extremamente próximos aos resultados apresentados por Albensoeder e Kuhlmann (2005), apenas com algumas pequenas discrepâncias aparecendo pontualmente em pontos de extremos. Deste modo, a excelente concordância dos resultados, mesmo para o caso de  $Re = 1000$  e de cavidades não-cúbicas, indica que as pequenas discrepâncias tendem a desaparecer com a utilização de malhas mais refinadas. Entretanto, devido ao aumento exponencial de custo computacional asso-

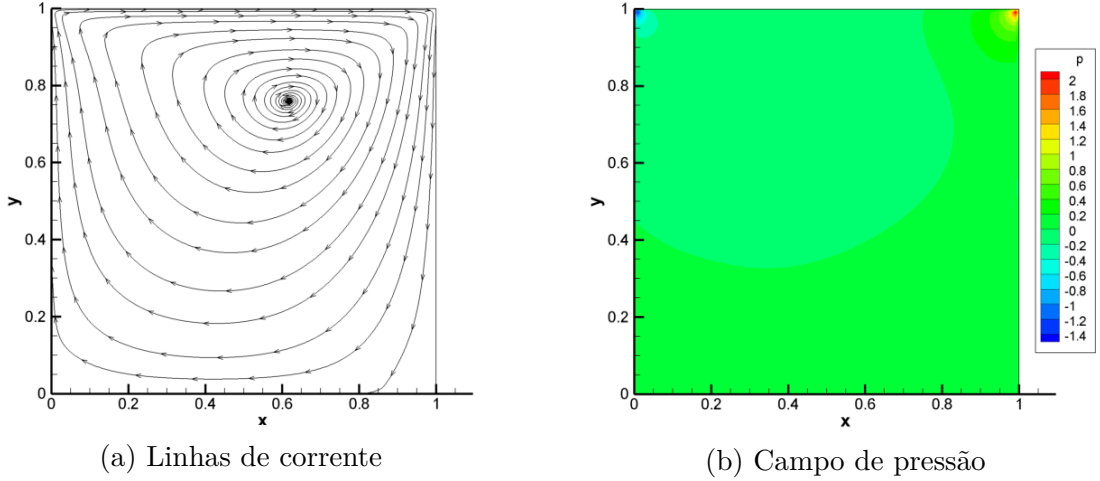


Figura 16 – Escoamento em uma cavidade cúbica com  $Re = 100$ . Plano  $z = 0,5$

ciado a um refinamento de malhas tridimensionais, a realização de simulações com malhas mais refinadas não foi possível.

Com isto, pode-se afirmar que a acurácia do código computacional desenvolvido se encontra validada, para números de Reynolds de até 1000, para diferentes dimensões de cavidade, e até mesmo para casos em que os valores de  $\Delta x$ ,  $\Delta y$  e  $\Delta z$  são diferentes. Além disso, a ordem 2 das discretizações temporal e espacial também foram comprovadas.

### 3.1.3 Análise do escoamento

De modo a permitir um melhor entendimento acerca do escoamento que se forma no interior da cavidade, esta seção apresenta visualizações dos escoamentos apresentados na seção anterior. Deste modo, as Figs. 16 e 17 apresentam o escoamento em uma cavidade cúbica com  $Re = 100$ , as Figs. 18 e 19 apresentam o escoamento em uma cavidade cúbica com  $Re = 1000$ , e a Fig. 20 apresenta o escoamento em uma cavidade com  $L_y = 2$ ,  $L_z = 1$ , e  $Re = 1000$ .

Para os dois primeiros casos, de cavidades cúbicas, os escoamentos são bastante semelhantes, porém com o escoamento de  $Re = 1000$  mais complexo, apresentando regiões secundárias de recirculação na região inferior da cavidade. Também é possível notar que o centro do vórtice principal muda de posição em função do número de Reynolds.

Com a visualização tridimensional das linhas de corrente, fica evidente que as principais componentes do escoamento estão situadas no plano  $xy$ , o que também é confirmado pelas pequenas magnitudes da componente  $w$  da velocidade (Fig. 15). Estas componentes do escoamento na direção  $z$  se manifestam como uma fuga do centro da cavidade a partir das regiões de recirculação secundárias, nos cantos inferiores, e em uma movimentação em direção ao centro do vórtice principal. Este comportamento está de acordo com os campos de pressão apresentados, em que a região próxima ao fundo da cavidade possui

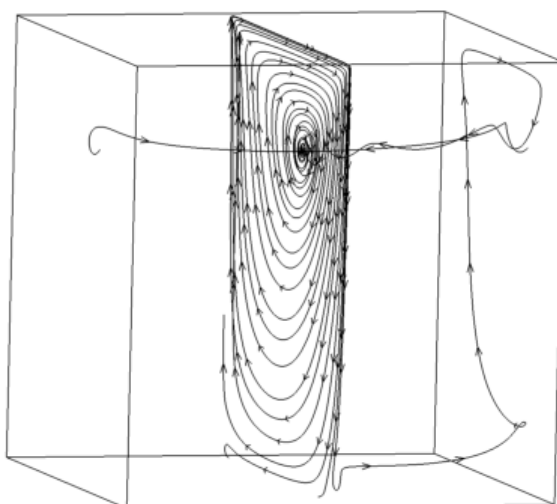


Figura 17 – Linhas de corrente de escoamento em uma cavidade cúbica com  $Re = 100$

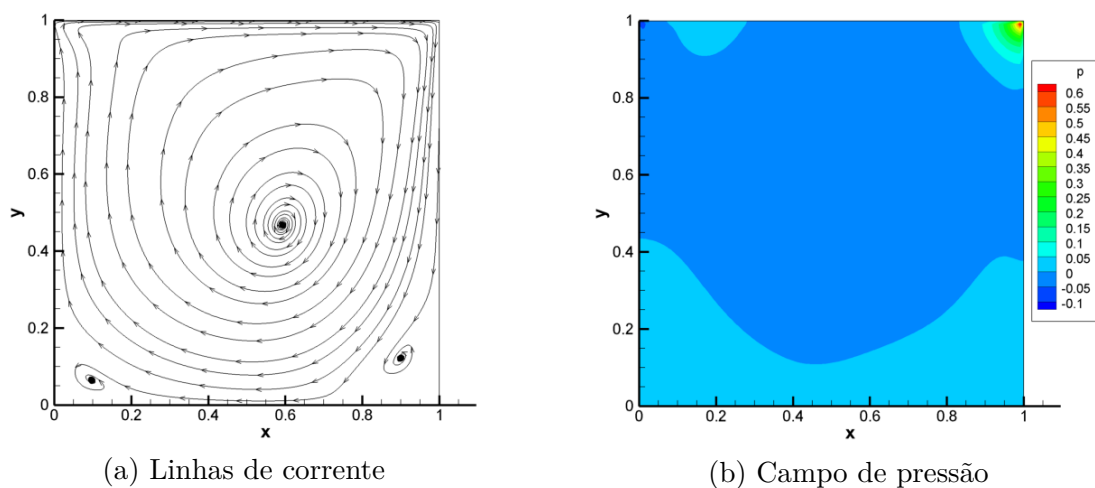


Figura 18 – Escoamento em uma cavidade cúbica com  $Re = 1000$ . Plano  $z = 0,5$

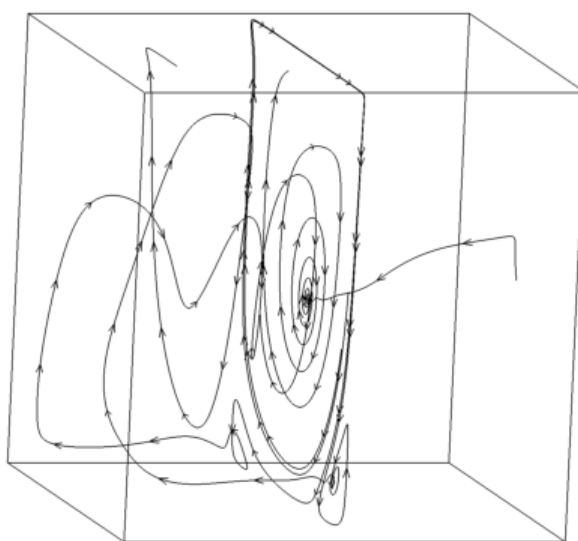


Figura 19 – Linhas de corrente de escoamento em uma cavidade cúbica com  $Re = 1000$

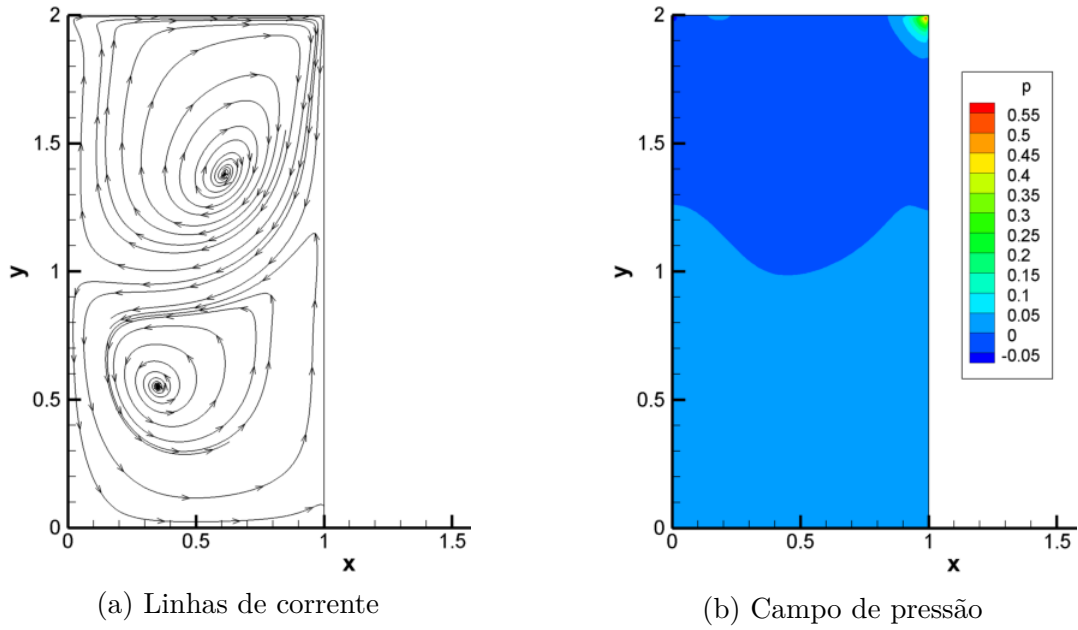


Figura 20 – Escoamento em uma cavidade com  $L_y = 2$ ,  $L_z = 1$ ,  $Re = 1000$

pressão maior que a região próxima ao centro do vórtice principal.

Para o terceiro caso, de uma cavidade mais alta, nota-se que o escoamento na metade superior da cavidade é muito semelhante ao escoamento em uma cavidade cúbica com mesmo número de Reynolds, não só nas linhas de corrente mas também no campo de pressão. Na metade inferior da cavidade, nota-se a formação de um segundo vórtice, de dimensões semelhantes às do vórtice principal, porém com uma intensidade menor (Figs. 13 e 14), o que se reflete na ausência de grandes variações significativas de pressão nessa região.

## 3.2 ESCOAMENTOS BIFÁSICOS DE CISALHAMENTO SIMPLES

### 3.2.1 Comparação com resultados disponíveis na literatura

Com o objetivo de validar o método de Level Set utilizado, foram realizadas simulações de deformação de gotas em cisalhamento simples, de modo a comparar os resultados obtidos com resultados disponíveis na literatura, bem como com a teoria de pequenas deformações desenvolvida por Taylor (1934). Os resultados da literatura utilizados são as simulações de integral de contorno realizadas por Rallison (1981) e Kennedy, Pozrikidis e Skalak (1994), e as simulações de volume de fluido realizadas por Li, Renardy e Renardy (2000).

As simulações foram realizadas para uma viscosidade igual dos dois fluidos, conforme limitação do método numérico desenvolvido neste trabalho, para escoamentos de Stokes ( $Re = 0$ ) com variados números de capilaridade. Foram utilizadas malhas de

$86 \times 86 \times 43$  para discretizar um domínio de  $8 \times 8 \times 4$ , com uma gota inicialmente esférica de raio unitário inserida no centro do domínio. Foi utilizado  $\Delta t = 5 \times 10^{-4}$ , com simulações indo até regime permanente. A perda de massa e os erros de  $|\nabla\phi|$  foram monitorados durante todas as simulações, de modo a garantir que se mantivessem em valores aceitáveis.

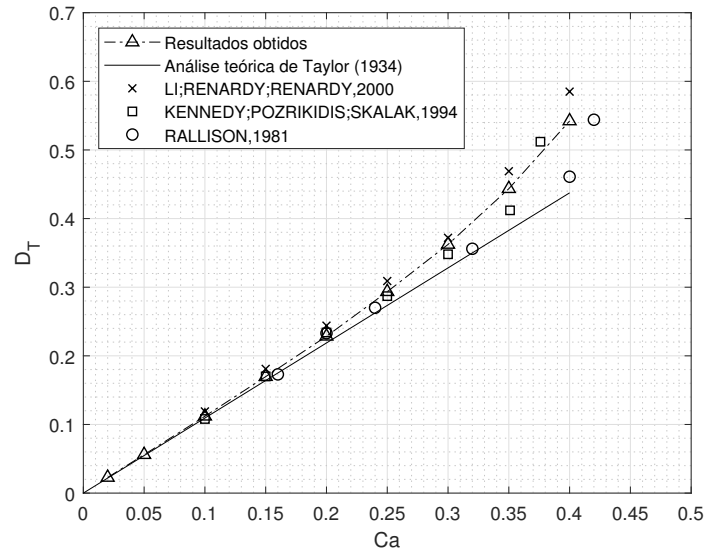


Figura 21 – Análise de deformação da gota

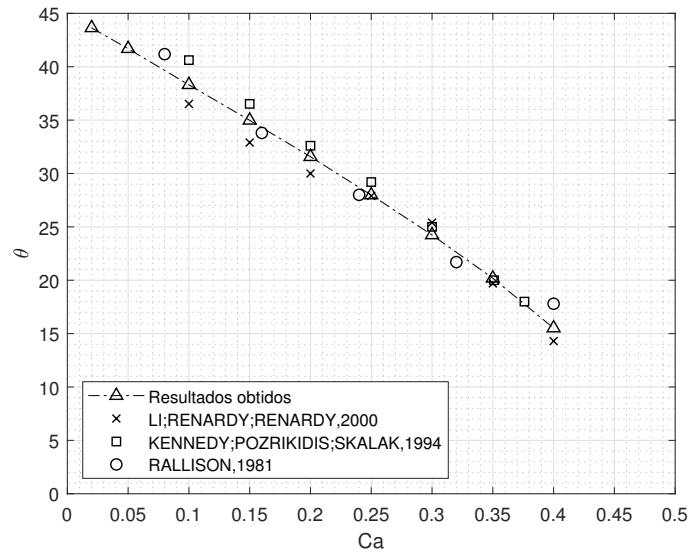


Figura 22 – Análise de inclinação da gota

As Figs. 21 e 22 apresentam os valores de deformação de Taylor e ângulo de inclinação da gota, respectivamente. Nota-se que há uma grande variação entre os resultados numéricos utilizados como referência, em que os resultados obtidos se situam dentro da região que abrange os resultados disponíveis na literatura. Também vale notar que os



resultados obtidos utilizam malhas pouco refinadas, de modo a reduzir o custo computacional das simulações, e possuem grande margem para melhoramento.

Por fim, também deseja-se validar a metodologia de cálculo da deformação de Taylor com base no tensor de inércia. Para tal, a Tab. 8 apresenta uma comparação entre os valores de deformação de Taylor calculados com base no tensor de inércia da gota, e os valores de deformação medidos a partir das imagens das respectivas gotas. Os valores calculados com base no tensor de inércia são os mesmos apresentados na Fig. 21.

$Ca$	0,02	0,05	0,1	0,15	0,2	0,25	0,3	0,35	0,4
Inércia	0,023	0,056	0,112	0,170	0,228	0,293	0,362	0,444	0,542
Imagem	0,027	0,047	0,113	0,174	0,229	0,297	0,367	0,442	0,539

Tabela 8 – Comparação entre formas de cálculo da deformação de Taylor.  $Re = 0$

Nota-se que os valores de deformação obtidos por ambos os métodos são extremamente próximos, especialmente para o caso de grandes deformações. Como para o caso de pequenas deformações os valores são muito sensíveis a erros de medição, e o formato da gota necessariamente será muito próximo à uma elipsoide, entende-se que as discrepâncias encontradas se devem a erros nos valores de deformação obtidos por medição de imagens, e que os valores de deformação calculados com base no tensor de inércia, nestes casos, é o mais correto. Deste modo, conclui-se que a metodologia proposta para o cálculo das deformações, com base no tensor de inércia, é adequada.

### 3.2.2 Análise do escoamento

De modo a permitir uma visualização do comportamento da gota sub um escoamento cisalhante simples, as Figs. 23, 24, 25 e 26 apresentam cortes no plano  $z = 2$  para alguns dos resultados apresentados na seção anterior, com  $Re = 0$  e variados números de  $Ca$ . Nas figuras de linhas de corrente, o traço preto corresponde ao contorno das gotas, e é possível observar que, conforme o número de Capilaridade aumenta, também aumentam a deformação e a inclinação da gota.

Também é possível observar que as linhas de corrente não atravessam o contorno da gota, comprovando que não há uma mistura entre os fluidos, já que o escoamento se encontra em regime permanente. Além disso, nota-se que há uma zona de recirculação no interior das gotas, indicando que, apesar de seu formato não variar, o fluido em seu interior está se movendo.

Já nas figuras dos campos de pressão, nota-se o salto de pressão causado pela tensão superficial, e que este se torna progressivamente menor conforme maiores valores de  $Ca$  tornam a tensão superficial menor. Também é possível observar que, conforme os

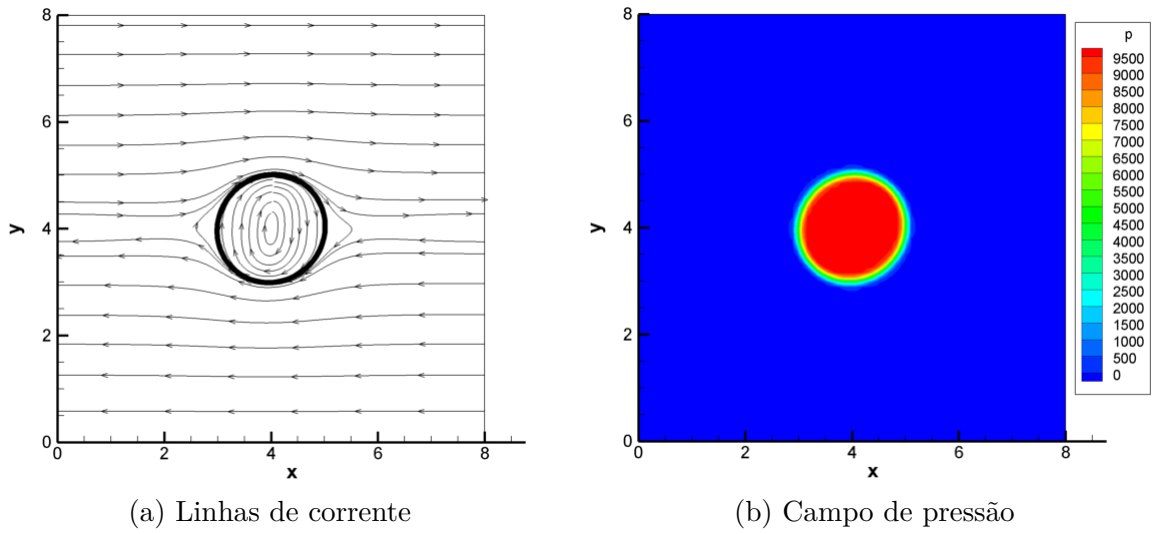


Figura 23 – Gota imersa em escoamento cisalhante simples,  $Re = 0$  e  $Ca = 0,02$

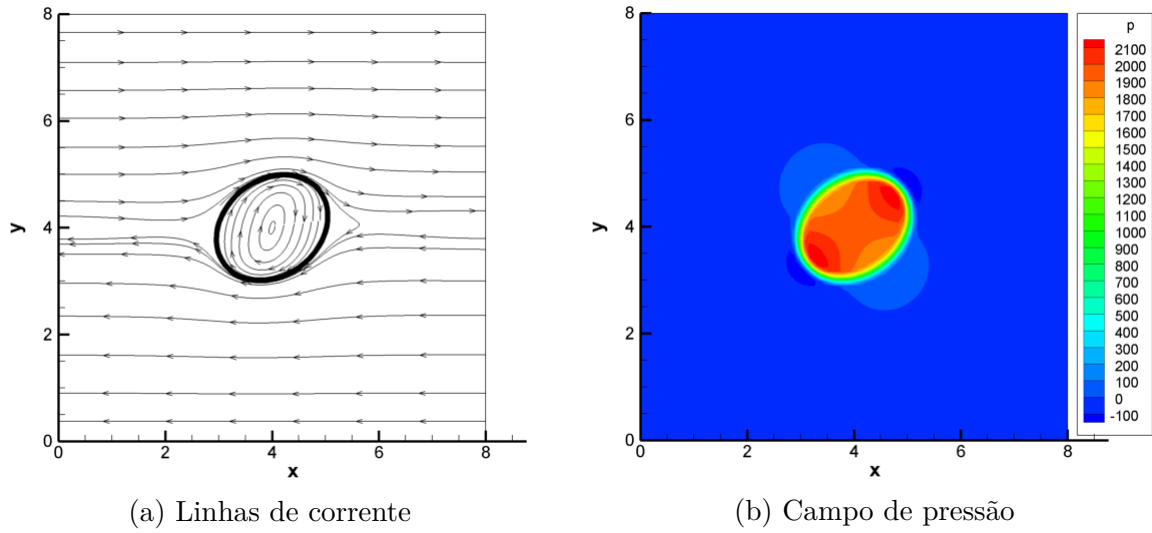


Figura 24 – Gota imersa em escoamento cisalhante simples,  $Re = 0$  e  $Ca = 0,1$

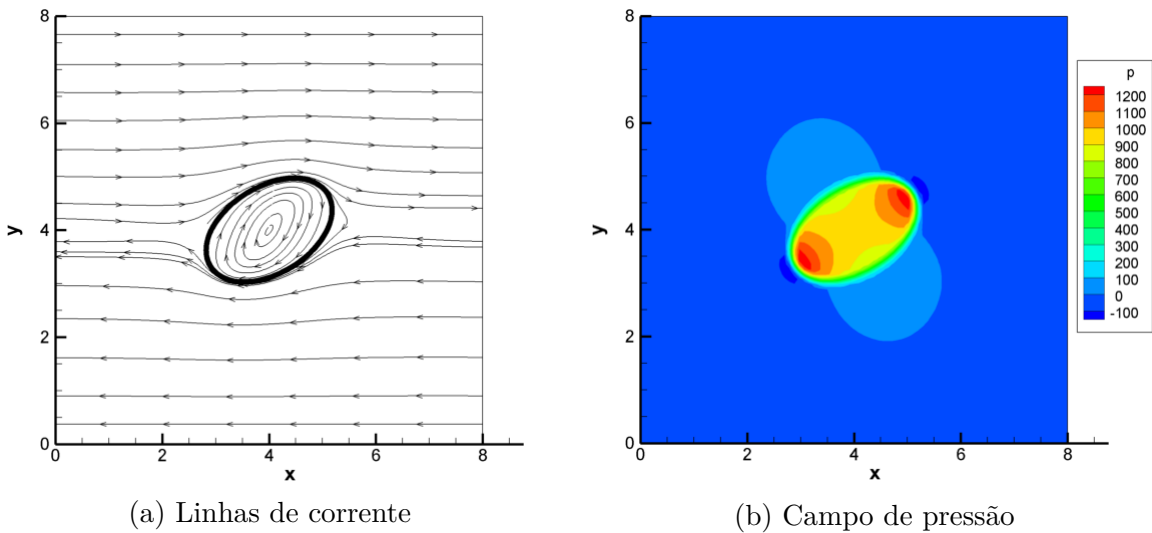


Figura 25 – Gota imersa em escoamento cisalhante simples,  $Re = 0$  e  $Ca = 0,2$

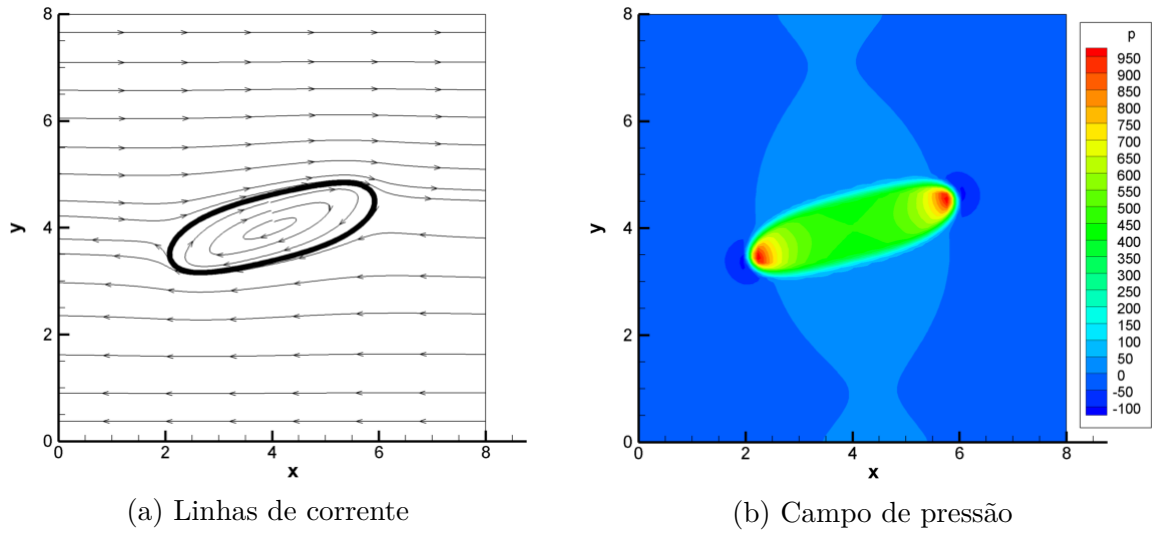


Figura 26 – Gota imersa em escoamento cisalhante simples,  $Re = 0$  e  $Ca = 0,4$

valores de  $Ca$  aumentam, mais significativas se tornam as variações de pressão longe da interface. Para o caso de  $Ca = 0,4$ , inclusive, é evidente a interação da gota com a parede, de modo que o ideal seria refazer esta simulação com um domínio ainda maior, para evitar qualquer alteração nos resultados causada pela proximidade da parede.

### 3.2.3 Ruptura de gotas

Para grandes valores de  $Ca$ , as tensões promovidas pelo escoamento se tornam suficientemente grandes, quando comparadas com a tensão superficial, e podem causar uma ruptura da gota. Este processo é responsável pela formação de emulsões, com gotas sendo rompidas até que suas dimensões se tornem extremamente pequenas, e a mistura de dois ou mais fluidos, em escala macroscópica, possa ser analisada como um único fluido, de propriedades distintas daquelas de seus fluidos constituintes.

Para o caso de cisalhamento simples, o processo de ruptura da gota é exemplificado pela Fig. 27, que apresenta o caso de  $Re = 0,1$  e  $Ca = 0,45$ . O domínio possui dimensões  $10 \times 4 \times 4$ , e foi utilizada uma malha de  $120 \times 48 \times 48$ , com  $\Delta t = 5 \times 10^{-4}$ .

Neste processo, observa-se que, quando a deformação da gota se torna suficientemente grande, ocorre um empescoamento em sua região central, que eventualmente se rompe. Como as gotas formadas pelo processo de ruptura possuem um raio menor, as tensões superficiais se tornam mais significativas, de modo que, após um dado número de rupturas, estas gotas não se romperão mais, sob o mesmo escoamento.

Usualmente, há a presença de pequenas gotas, denominadas satélites, na região em que se situava o pescoço. Na simulação aqui apresentada, estas gotas não estão presentes, pois a perda de massa se mostrou extremamente grande, com o volume final correspondendo a apenas cerca de 60% do volume inicial, e causou o desaparecimento destas gotas.

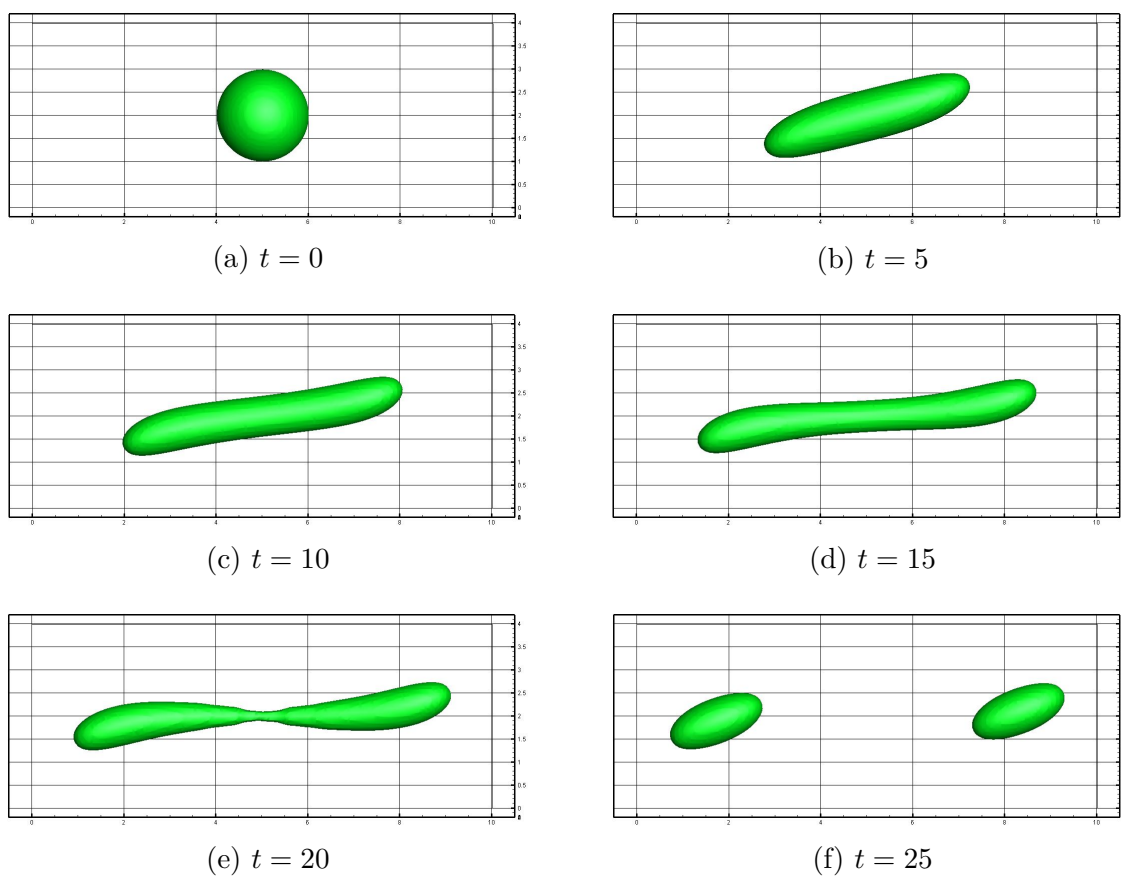


Figura 27 – Ruptura de gota sob escoamento cisalhante simples.  $Re = 0,1$ ,  $Ca = 0,45$

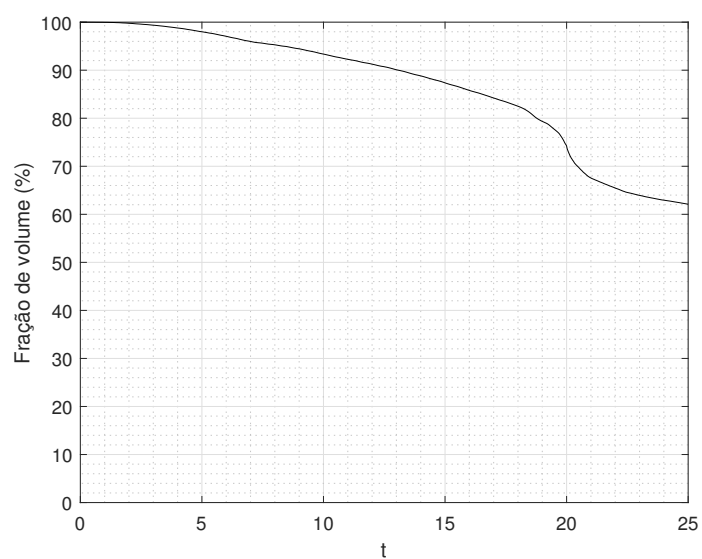


Figura 28 – Análise da perda de volume da gota

A evolução desta perda de massa está disposta na Fig. 28, em que se nota uma perda bastante abrupta na região próxima a  $t = 20$ , quando ocorre a ruptura.

Deste modo, os resultados apresentados nesta seção não são precisos, devido à falta de refinamento da malha utilizada, e servem apenas para demonstrar a capacidade do método de Level Set utilizado em lidar com fenômenos de ruptura de gotas. Para que a perda de massa fosse mantida em níveis aceitáveis, seria necessário um grande refinamento da malha, o que tornaria esta simulação inviável sem um grande aumento na capacidade de processamento computacional disponível.

## 4 CONCLUSÕES E TRABALHOS FUTUROS

Foi desenvolvido um método numérico capaz de realizar simulações de escoamentos bifásicos tridimensionais em regime laminar. Este método foi validado, tanto para escoamentos monofásicos quanto para escoamentos bifásicos, sendo comprovadamente de segunda ordem tanto na discretização espacial quanto na discretização temporal. Também foram comprovadas a capacidade deste método de lidar com fenômenos de ruptura de gotas, e a validade de se calcular a deformação de gotas com base na hipótese de que elas são aproximadamente elipsoides.

Para o caso de escoamentos monofásicos, foram comparados resultados obtidos com o método desenvolvido e resultados de referência disponíveis na literatura para o problema da cavidade, em que os resultados obtidos estavam extremamente próximos aos resultados de referência. Para o caso de escoamentos bifásicos, foi realizada uma validação com base na deformação de Taylor e inclinação de uma gota imersa em um escoamento cisalhante simples, com comparações realizadas com resultados disponíveis na literatura, obtidos por diferentes métodos numéricos, bem como com aproximações teóricas. Os resultados obtidos estão de acordo com o esperado, tendo-se em vista que há uma certa discrepância mesmo entre os resultados disponíveis na literatura.

Também foi realizada uma análise dos escoamentos apresentados, de modo a se observar o comportamento dos fluidos, bem como a influência de variações dos números de Reynolds e de Capilaridade. Por fim, foi realizada uma análise de viabilidade da utilização de variáveis do tipo *single-precision* e *double-precision*, com base nas velocidades de processamento e na precisão dos resultados obtidos (Anexo B).

Em desenvolvimentos futuros deste código, deseja-se, primeiramente, investigar a possibilidade de utilização da plataforma CUDA, ou da plataforma OpenCL, para realizar simulações com a utilização de GPUs, o que permitiria um grande aumento na capacidade de processamento disponível. Caso esta implementação seja bem sucedida, são esperados ganhos de rapidez de uma ordem de magnitude. Métodos melhores de solução dos sistemas lineares também serão estudados, como métodos do tipo *Fast Fourier Solvers* e métodos de *multigrid*. Devido ao alto custo computacional associado a simulações tridimensionais, melhorias de desempenho do código são extremamente importantes. Também há a possibilidade de implementação de metodologias que promovam uma melhor conservação de massa do Level Set, o que permitiria a realização de simulações mais longas sem a necessidade de utilizar malhas excessivamente finas.

Posteriormente, também deseja-se alterar o método de projeção utilizado, de forma a permitir simulações em que as viscosidades (e possivelmente densidades) dos fluidos sejam diferentes, o que tornaria possível uma ampla gama de novas simulações a serem realizadas, incluindo novos métodos de validação, e uma maior quantidade de resultados para os métodos de validação já utilizados. Além disso, também é considerada a hipótese de se implementar metodologias para simulação de fluidos e campos magnéticos, de modo a permitir o estudo de emulsões com fluidos magneto-reológicos.

# Referências

- ADALSTEINSSON, D.; SETHIAN, J. A. A fast level set method for propagating interfaces. *Journal of computational physics*, Elsevier, v. 118, n. 2, p. 269–277, 1995. Citado na página 20.
- ALBENSOEDER, S.; KUHLMANN, H. Accurate three-dimensional lid-driven cavity flow. *Journal of Computational Physics*, v. 206, n. 2, p. 536 – 558, 2005. ISSN 0021-9991. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0021999105000033>>. Citado 6 vezes nas páginas 15, 20, 55, 57, 58 e 59.
- BENTLEY, B.; LEAL, L. G. An experimental investigation of drop deformation and breakup in steady, two-dimensional linear flows. *Journal of Fluid Mechanics*, Cambridge University Press, v. 167, p. 241–283, 1986. Citado na página 17.
- BOLZ, J. et al. Sparse matrix solvers on the gpu: conjugate gradients and multigrid. *ACM transactions on graphics (TOG)*, ACM, v. 22, n. 3, p. 917–924, 2003. Citado na página 21.
- BROWN, D. L.; CORTEZ, R.; MINION, M. L. Accurate projection methods for the incompressible navier–stokes equations. *Journal of computational physics*, Elsevier, v. 168, n. 2, p. 464–499, 2001. Citado 3 vezes nas páginas 19, 23 e 24.
- CHANG, Y.-C. et al. A level set formulation of eulerian interface capturing methods for incompressible fluid flows. *Journal of computational Physics*, Elsevier, v. 124, n. 2, p. 449–464, 1996. Citado na página 19.
- ENRIGHT, D. et al. A hybrid particle level set method for improved interface capturing. *Journal of Computational physics*, Elsevier, v. 183, n. 1, p. 83–116, 2002. Citado na página 19.
- FERZIGER, J. H.; PERIC, M. *Computational methods for fluid dynamics*. [S.l.]: Springer Science & Business Media, 2012. Citado na página 35.
- GHIA, U.; GHIA, K. N.; SHIN, C. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of computational physics*, Elsevier, v. 48, n. 3, p. 387–411, 1982. Citado na página 20.
- HIRT, C. W.; NICHOLS, B. D. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of computational physics*, Elsevier, v. 39, n. 1, p. 201–225, 1981. Citado na página 18.
- JIANG, G.-S.; SHU, C.-W. Efficient implementation of weighted eno schemes. *Journal of computational physics*, Elsevier, v. 126, n. 1, p. 202–228, 1996. Citado na página 37.
- KENNEDY, M.; POZRIKIDIS, C.; SKALAK, R. Motion and deformation of liquid drops, and the rheology of dilute emulsions in simple shear flow. *Computers & fluids*, Elsevier, v. 23, n. 2, p. 251–278, 1994. Citado 2 vezes nas páginas 18 e 62.



- KIM, J.; MOIN, P. Application of a fractional-step method to incompressible navier-stokes equations. *Journal of Computational Physics*, v. 59, n. 2, p. 308 – 323, 1985. ISSN 0021-9991. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0021999185901482>>. Citado 4 vezes nas páginas 19, 22, 23 e 34.
- KOPP, J. Efficient numerical diagonalization of hermitian  $3 \times 3$  matrices. *International Journal of Modern Physics C*, World Scientific, v. 19, n. 03, p. 523–548, 2008. Citado na página 50.
- LEE, C.; DOLBOW, J.; MUCHA, P. J. A narrow-band gradient-augmented level set method for multiphase incompressible flow. *Journal of Computational Physics*, Elsevier, v. 273, p. 12–37, 2014. Citado na página 18.
- LI, C. et al. Level set evolution without re-initialization: a new variational formulation. In: IEEE. *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*. [S.l.], 2005. v. 1, p. 430–436. Citado na página 19.
- LI, J.; RENARDY, Y. Y.; RENARDY, M. Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method. *Physics of Fluids*, AIP, v. 12, n. 2, p. 269–282, 2000. Citado 2 vezes nas páginas 18 e 62.
- MIN, C. On reinitializing level set functions. *Journal of computational physics*, Elsevier, v. 229, n. 8, p. 2764–2772, 2010. Citado na página 19.
- OLSSON, E.; KREISS, G. A conservative level set method for two phase flow. *Journal of computational physics*, Elsevier, v. 210, n. 1, p. 225–246, 2005. Citado na página 20.
- OSHER, S.; FEDKIW, R. *Level set methods and dynamic implicit surfaces*. [S.l.]: Springer Science & Business Media, 2006. v. 153. Citado 3 vezes nas páginas 28, 37 e 43.
- OSHER, S.; FEDKIW, R. P. Level set methods: an overview and some recent results. *Journal of Computational physics*, Elsevier, v. 169, n. 2, p. 463–502, 2001. Citado na página 18.
- OSHER, S.; SETHIAN, J. A. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, Elsevier, v. 79, n. 1, p. 12–49, 1988. Citado na página 18.
- PENG, D. et al. A pde-based fast local level set method. *Journal of computational physics*, Elsevier, v. 155, n. 2, p. 410–438, 1999. Citado 2 vezes nas páginas 20 e 31.
- RALLISON, J. A numerical study of the deformation and burst of a viscous drop in general shear flows. *Journal of Fluid Mechanics*, Cambridge University Press, v. 109, p. 465–482, 1981. Citado 2 vezes nas páginas 18 e 62.
- RUMSCHEIDT, F.-D.; MASON, S. Particle motions in sheared suspensions xii. deformation and burst of fluid drops in shear and hyperbolic flow. *Journal of Colloid Science*, Elsevier, v. 16, n. 3, p. 238–261, 1961. Citado na página 17.
- SAAD, Y. *Iterative methods for sparse linear systems*. [S.l.]: siam, 2003. v. 82. Citado 2 vezes nas páginas 35 e 48.

- SHEWCHUK, J. R. et al. *An introduction to the conjugate gradient method without the agonizing pain*. [S.l.]: Carnegie-Mellon University. Department of Computer Science, 1994. Citado 2 vezes nas páginas 7 e 48.
- SHU, C.-W.; OSHER, S. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, Elsevier, v. 77, n. 2, p. 439–471, 1988. Citado na página 32.
- SUSSMAN, M.; FATEMI, E. An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal on scientific computing*, SIAM, v. 20, n. 4, p. 1165–1191, 1999. Citado na página 19.
- SUSSMAN, M. et al. An improved level set method for incompressible two-phase flows. *Computers and Fluids*, v. 27, n. 5, p. 663 – 680, 1998. ISSN 0045-7930. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0045793097000534>>. Citado 3 vezes nas páginas 19, 27 e 28.
- SUSSMAN, M.; SMEREKA, P.; OSHER, S. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational physics*, Elsevier, v. 114, n. 1, p. 146–159, 1994. Citado 3 vezes nas páginas 19, 20 e 30.
- TAYLOR, G. I. The viscosity of a fluid containing small drops of another fluid. *Proc. R. Soc. Lond. A*, The Royal Society, v. 138, n. 834, p. 41–48, 1932. Citado na página 17.
- TAYLOR, G. I. The formation of emulsions in definable fields of flow. *Proc. R. Soc. Lond. A*, The Royal Society, v. 146, n. 858, p. 501–523, 1934. Citado 3 vezes nas páginas 17, 51 e 62.
- TORZA, S.; COX, R.; MASON, S. Particle motions in sheared suspensions xxvii. transient and steady deformation and burst of liquid drops. *Journal of Colloid and Interface science*, Elsevier, v. 38, n. 2, p. 395–411, 1972. Citado na página 17.
- TSITSIKLIS, J. N. Efficient algorithms for globally optimal trajectories. *IEEE Transactions on Automatic Control*, IEEE, v. 40, n. 9, p. 1528–1538, 1995. Citado na página 19.
- WEINAN, E.; LIU, J.-G. et al. Gauge method for viscous incompressible flows. *Communications in Mathematical Sciences*, International Press of Boston, v. 1, n. 2, p. 317–332, 2003. Citado na página 25.
- YANG, S.; GOBBERT, M. K. The optimal relaxation parameter for the sor method applied to the poisson equation in any space dimensions. *Applied Mathematics Letters*, Elsevier, v. 22, n. 3, p. 325–331, 2009. Citado na página 48.

# ANEXO A – PARÂMETROS DOS SISTEMAS LINEARES NA FORMA DE ESTÊNCEL

Variável	a	b	c	d	e	f	g
$u^*$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$	$1 + \frac{\Delta t}{Re\Delta x^2} + \frac{\Delta t}{Re\Delta y^2} + \frac{\Delta t}{Re\Delta z^2}$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$
$v^*$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$	$1 + \frac{\Delta t}{Re\Delta x^2} + \frac{\Delta t}{Re\Delta y^2} + \frac{\Delta t}{Re\Delta z^2}$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$
$w^*$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$	$1 + \frac{\Delta t}{Re\Delta x^2} + \frac{\Delta t}{Re\Delta y^2} + \frac{\Delta t}{Re\Delta z^2}$	$-\frac{\Delta t}{2Re\Delta x^2}$	$-\frac{\Delta t}{2Re\Delta y^2}$	$-\frac{\Delta t}{2Re\Delta z^2}$
$\chi$	$\frac{1}{\Delta x^2}$	$\frac{1}{\Delta y^2}$	$\frac{1}{\Delta z^2}$	$-\frac{2}{\Delta x^2} - \frac{2}{\Delta y^2} - \frac{2}{\Delta z^2}$	$\frac{1}{\Delta x^2}$	$\frac{1}{\Delta y^2}$	$\frac{1}{\Delta z^2}$

Tabela 9 – Parâmetros dos sistemas lineares na forma do estêncil

## ANEXO B – ANÁLISE DA INFLUÊNCIA DE UTILIZAÇÃO DE VARIÁVEIS DE PRECISÃO SIMPLES OU DUPLA

Em todas as simulações apresentadas neste relatório, foram utilizadas variáveis do tipo *double-precision*, de 8 bytes (64 bits), resultando em uma média de 16 algarismos de precisão. Como forma de comparação, também foram realizadas algumas simulações com variáveis do tipo *single-precision*, de 4 bytes (32 bits), resultando em uma média de 7 algarismos de precisão. Devido ao menor tamanho dos algarismos, o custo computacional de realizar as operações é reduzido, fazendo com que simulações com variáveis do tipo *single-precision* sejam muito mais rápidas que simulações equivalentes com variáveis do tipo *double-precision*.

Para comparar os resultados, foram realizadas algumas simulações com parâmetros idênticos a simulações já apresentadas previamente, alterando somente o nível de precisão das variáveis, e a tolerância utilizada para o algoritmo de gradientes conjugados. Com a redução do nível de precisão das variáveis, a utilização de tolerâncias muito pequenas ( $10^{-10}$  nas simulações com variáveis *double-precision*) se mostrou impossível para os casos de escoamentos bifásicos. As simulações realizadas com variáveis do tipo *single-precision* foram dois casos de cisalhamento simples bifásico, um com  $Re = 0$  e  $Ca = 0,2$ , semelhante ao apresentado na seção 3.2.1, utilizando uma tolerância de  $5 \times 10^{-5}$ , e um com  $Re = 0,1$  e  $Ca = 0,45$ , semelhante ao apresentado na seção 3.2.3, com tolerâncias de  $5 \times 10^{-5}$  e  $10^{-3}$ , além de um caso do problema da cavidade monofásico, para uma cavidade cúbica com  $Re = 1000$ , semelhante ao apresentado na seção 3.1.2, com uma tolerância de  $10^{-3}$ .

Para o primeiro caso, de escoamento cisalhante simples com  $Re = 0$  e  $Ca = 0,2$ , as Figs. 29, 30 e 31 apresentam gráficos de volume, ângulo de inclinação e deformação de Taylor, respectivamente, em função do número de iterações realizadas. Neste caso, a tolerância de  $5 \times 10^{-5}$  utilizada é a menor possível, já que o algoritmo de gradientes conjugados não conseguiu convergir para menores valores de tolerância. Nota-se que não há uma diferença significativa entre os resultados obtidos com variáveis do tipo *single-precision* e os resultados com variáveis do tipo *double-precision*. Neste caso, a simulação com variáveis do tipo *single-precision* foi mais de 4 vezes mais rápida que a simulação com variáveis do tipo *double-precision*, porém é importante atentar para o fato que, devido à diferença nas tolerâncias utilizadas, este ganho de desempenho não se deve unicamente à alteração do tipo de variável.

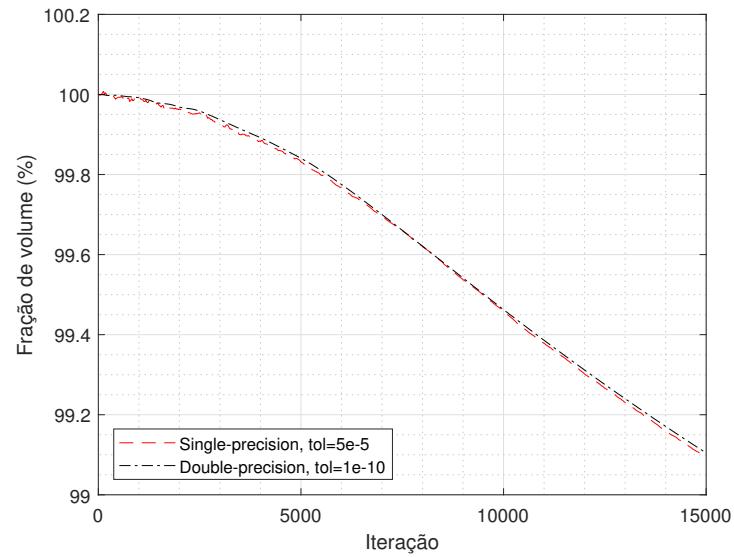


Figura 29 – Comparação de perdas de volume para  $Re = 0$  e  $Ca = 0,2$

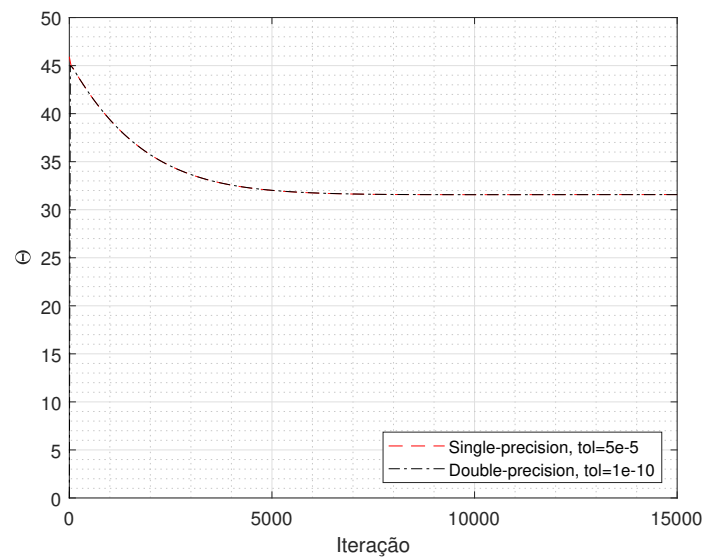


Figura 30 – Comparação de ângulos de inclinação para  $Re = 0$  e  $Ca = 0,2$

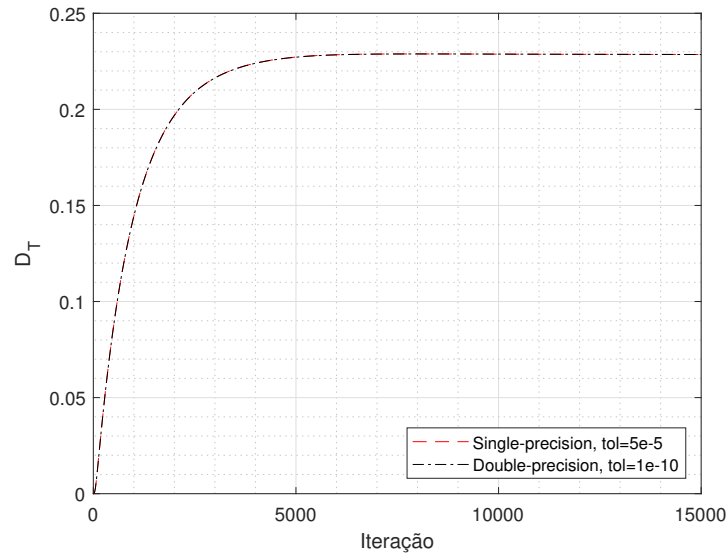


Figura 31 – Comparação de deformações de Taylor para  $Re = 0$  e  $Ca = 0,2$

Para o segundo caso, de escoamento cisalhante simples com  $Re = 0,1$  e  $Ca = 0,45$ , os gráficos de volume, ângulo de inclinação e deformação de Taylor estão dispostos nas Figs. 32, 33 e 34, respectivamente. Todas as simulações foram realizadas com uma malha de  $120 \times 48 \times 48$ , com exceção de uma das simulações com tolerância de  $10^{-3}$ , na qual foi utilizada uma malha de  $125 \times 50 \times 50$ . Os tempos de simulação, novamente, foram mais de 4 vezes menores para o caso de variáveis do tipo *single-precision*. Novamente, é importante atentar para o fato que, devido à diferença nas tolerâncias utilizadas, este ganho de desempenho não se deve unicamente à alteração do tipo de variável.

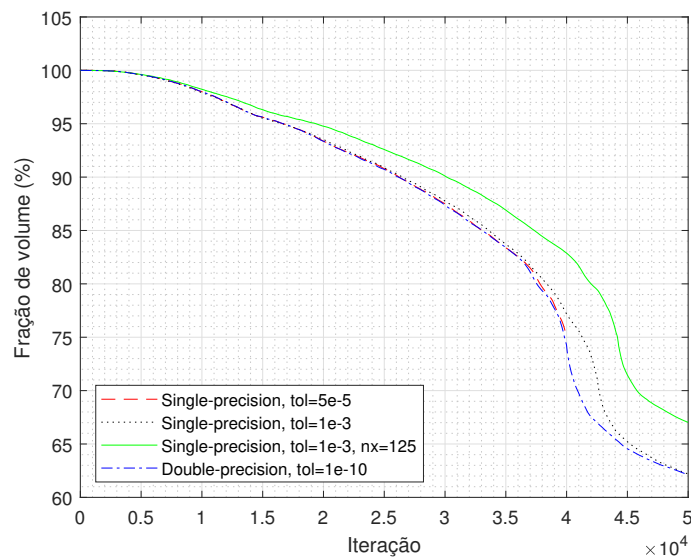


Figura 32 – Comparação de perdas de volume para  $Re = 0,1$  e  $Ca = 0,45$

Nota-se que, novamente, os resultados obtidos para variáveis do tipo *single-precision* são extremamente próximos, porém somente até o início do processo de ruptura. Neste

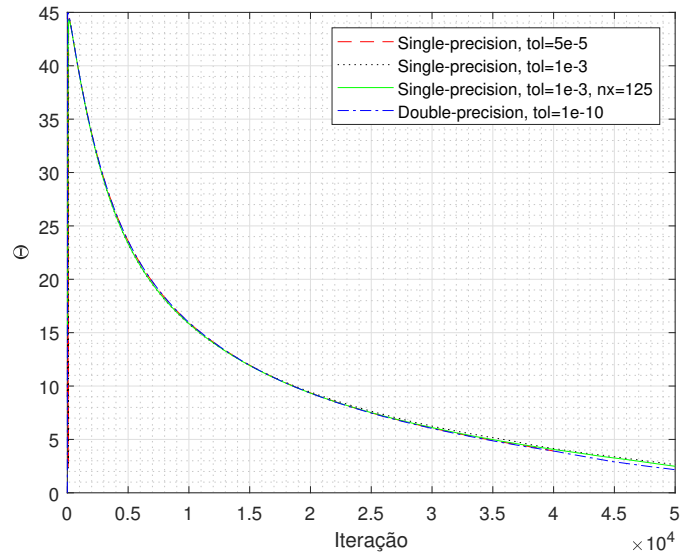


Figura 33 – Comparação de ângulos de inclinação para  $Re = 0, 1$  e  $Ca = 0, 45$

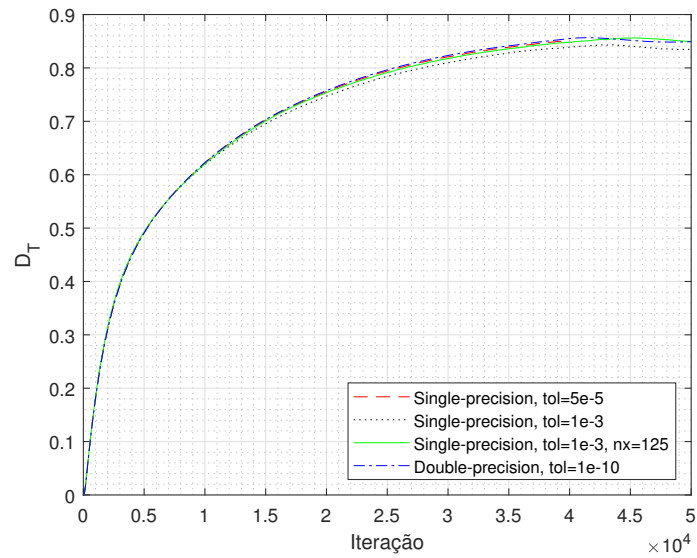


Figura 34 – Comparação de deformações de Taylor para  $Re = 0, 1$  e  $Ca = 0, 45$



ponto, o algoritmo de gradientes conjugados não conseguiu convergir para uma tolerância de  $5 \times 10^{-5}$ , e foi necessário reduzir esta tolerância para  $10^{-3}$ . Com esta tolerância reduzida, a deformação da gota foi ligeiramente menor, fazendo com que a ruptura ocorresse depois, como pode ser observado no gráfico de perda de volume, com a queda repentina de volume ocorrendo depois. Para o caso da malha ligeiramente mais refinada, de  $125 \times 50 \times 50$ , a perda de volume foi bastante reduzida, de modo geral, o que se deve à malha mais refinada. Como a parte de cálculo do Level Set possui uma discretização de quinta ordem para a discretização espacial, esta pequena alteração na malha foi suficiente para gerar alterações perceptíveis. Nota-se que a ruptura demora mais a ocorrer, o que indica que o processo de ruptura muito provavelmente foi acelerado pela perda de volume.

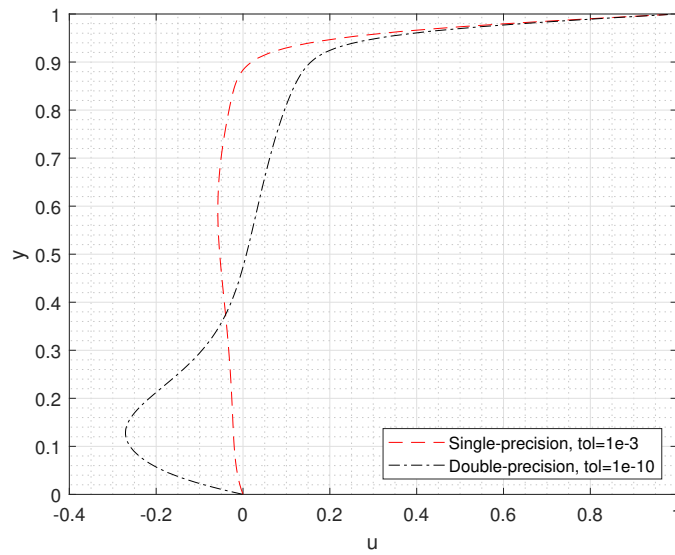


Figura 35 – Comparação de perfis da velocidade  $u$  para  $x = 0,5$  e  $z = 0,5$ .  $Re = 1000$

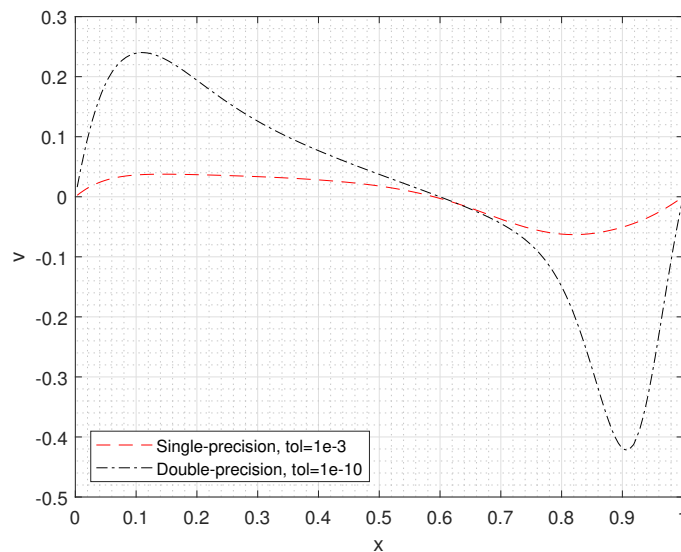


Figura 36 – Comparação de perfis da velocidade  $v$  para  $y = 0,5$  e  $z = 0,5$ .  $Re = 1000$

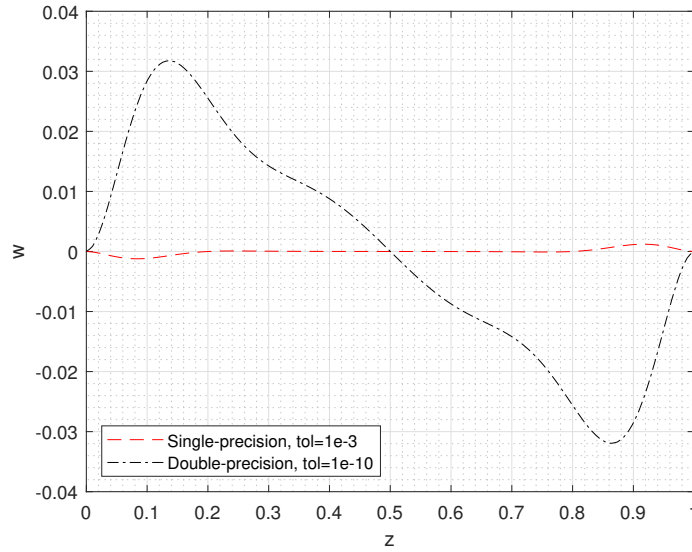


Figura 37 – Comparação de perfis da velocidade  $w$  para  $x = 0,5$  e  $y = 0,5$ .  $Re = 1000$

Para o terceiro caso, de escoamento monofásico em uma cavidade cúbica com  $Re = 1000$ , as Figs. 35, 36 e 37 apresentam perfis das componentes de velocidade  $u$ ,  $v$  e  $w$ , respectivamente, no instante  $t = 60$ . Neste caso, foi utilizada uma tolerância de  $10^{-3}$  para a simulação com variáveis do tipo *single-precision*, fazendo com que esta simulação fosse mais de 4 vezes mais rápida que a simulação realizada com variáveis do tipo *double-precision*. Entretanto, como evidenciado pelos gráficos, esta redução na precisão gerou erros inaceitáveis. Além disso, grande parte do ganho de rapidez da simulação se deve ao aumento da tolerância, e não ao tipo de variável utilizado.

Neste caso de escoamento monofásico na cavidade, foi possível realizar uma simulação com variáveis do tipo *single-precision* e uma tolerância de  $10^{-10}$ , a mesma utilizada para as simulações com variáveis do tipo *double-precision*, porém isto fez com que o ganho de rapidez da simulação fosse bastante reduzido, com a simulação com variáveis do tipo *single-precision* sendo somente cerca de 2 a 3 vezes mais rápida. Neste caso, porém, este ganho de rapidez se deve unicamente ao tipo de variável utilizado. Os resultados desta simulação, para o instante  $t = 30$ , estão dispostos nas Figs. 38, 39 e 40, com perfis das componentes de velocidade  $u$ ,  $v$  e  $w$ , respectivamente, juntamente com perfis calculados com variáveis do tipo *double-precision*, neste mesmo instante de tempo.

Nestas figuras, nota-se que os resultados obtidos com variáveis do tipo *single-precision* foram idênticos aos obtidos com variáveis do tipo *double-precision*. Deste modo, fica confirmado que os grandes erros apresentados nas Figs. 35, 36 e 37 se deve unicamente ao alto valor utilizado para a tolerância, e não ao nível de precisão utilizado nas variáveis.

Em conclusão, a utilização de variáveis do tipo *single-precision* se mostra ideal para casos em que é possível adotar valores adequadamente pequenos para a tolerância,

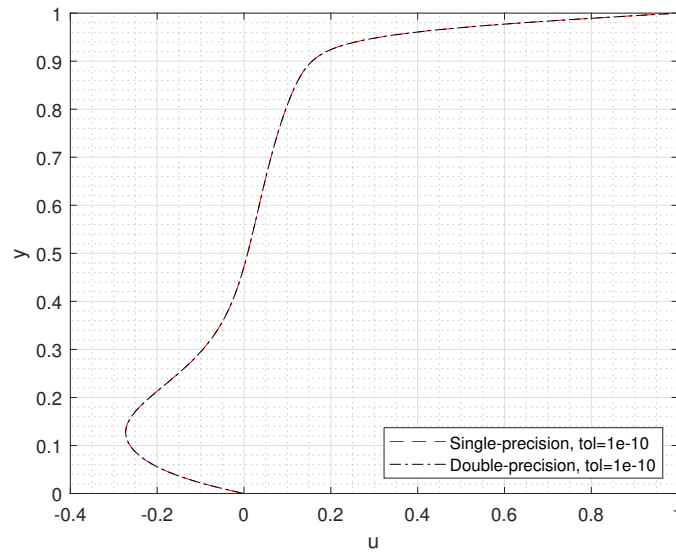


Figura 38 – Comparação de perfis da velocidade  $u$  para  $x = 0,5$  e  $z = 0,5$ .  $Re = 1000$ ,  $t = 30$

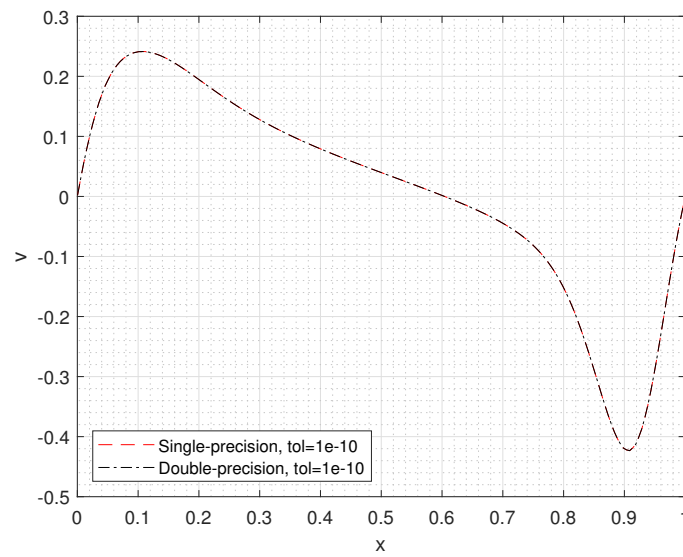


Figura 39 – Comparação de perfis da velocidade  $v$  para  $y = 0,5$  e  $z = 0,5$ .  $Re = 1000$ ,  $t = 30$

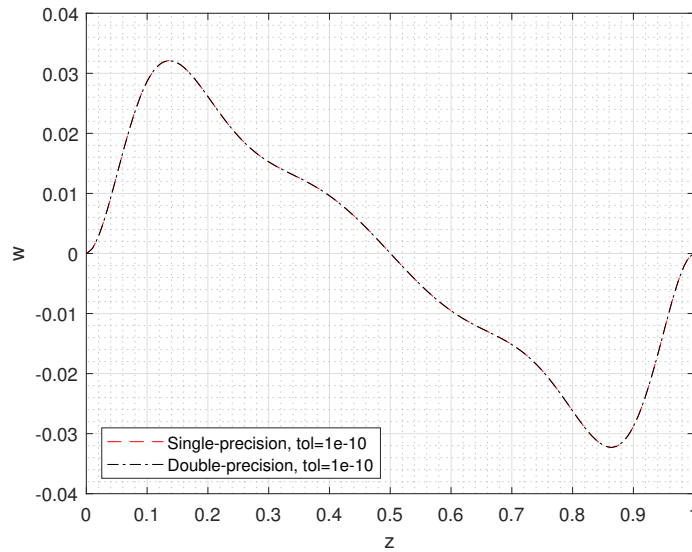


Figura 40 – Comparação de perfis da velocidade  $w$  para  $x = 0,5$  e  $y = 0,5$ .  $Re = 1000$ ,  $t = 30$

já que, nestes casos, as alterações nos resultados são praticamente inexistentes, e os ganhos de rapidez, extremamente relevantes, não trazem consequências negativas. Estes ganhos de rapidez, inclusive, permitem a utilização de malhas mais refinadas, o que aumentaria a precisão dos resultados.

Para os casos em que for necessário aumentar o valor da tolerância para se conseguir uma convergência do método de gradientes conjugados, como ocorreu para o caso de simulações de escoamentos bifásicos, entretanto, deve-se tomar extremo cuidado, já que este aumento da tolerância, caso seja muito grande, pode gerar erros inaceitáveis. Para estes casos, portanto, a utilização de variáveis do tipo *double-precision* pode ser mais recomendada.

Por fim, a escolha do tipo de variável a ser utilizada também deve levar em consideração o tipo de *hardware* utilizado, já que diferentes arquiteturas de processador possuem diferentes desempenhos relativos entre operações com números de 32 e de 64 bits. Na CPU utilizada para realizar as simulações aqui apresentadas, por exemplo, o desempenho relativo é próximo de 1:2 (como evidenciado pelas simulações com tolerâncias iguais), isto é, operações com números de 32 bits são cerca de 2 vezes mais rápidas que operações com números de 64 bits, o que está muito próximo do máximo teórico de 1:2 - deste modo, os ganhos de rapidez muito superiores a 2 muito provavelmente se devem à redução no número de operações causada pelo aumento do valor da tolerância. Para outras arquiteturas de processamento, entretanto, o desempenho relativo pode variar muito mais, com algumas GPUs modernas, por exemplo, tendo um desempenho relativo de 1:32, isto é, operações com números do tipo *double-precision* são 32 vezes mais lentas que operações com números do tipo *single-precision*.